Subspace Representation Learning for Sparse Linear Arrays to Localize More Sources Than Sensors: A Deep Learning Methodology

Kuan-Lin Chen, *Member, IEEE* and Bhaskar D. Rao, *Life Fellow, IEEE*

***Abstract*—Localizing more sources than sensors with a sparse linear array (SLA) has long relied on minimizing a distance between two covariance matrices and recent algorithms often utilize semidefinite programming (SDP). Although deep neural network (DNN)-based methods offer new alternatives, they still depend on covariance matrix fitting. In this paper, we develop a novel methodology that estimates the co-array subspaces from a sample covariance for SLAs. Our methodology trains a DNN to learn signal and noise subspace representations that are invariant to the selection of bases. To learn such representations, we propose loss functions that gauge the separation between the desired and the estimated subspace. In particular, we propose losses that measure the length of the shortest path between subspaces viewed on a union of Grassmannians, and prove that it is possible for a DNN to approximate signal subspaces. The computation of learning subspaces of different dimensions is accelerated by a new batch sampling strategy called consistent rank sampling. The methodology is robust to array imperfections due to its geometry-agnostic and data-driven nature. In addition, we propose a fully end-to-end gridless approach that directly learns angles to study the possibility of bypassing subspace methods. Numerical results show that learning such subspace representations is more beneficial than learning covariances or angles. It outperforms conventional SDP-based methods such as the sparse and parametric approach (SPA) and existing DNN-based covariance reconstruction methods for a wide range of signal-to-noise ratios (SNRs), snapshots, and source numbers for both perfect and imperfect arrays.**

***Index Terms*—Neural networks, representation learning, subspaces, sparse linear arrays, direction-of-arrival estimation.**

I. Introduction

DIRECTION-of-arrival (DoA) estimation is one of the fundamental problems in array processing, providing the direction information of sources to many applications such as hearing aids [1], wireless communications [2], and sonar systems [3]. When a sufficiently large number of array measurements or *snapshots* are available, most approaches estimate a spatial covariance matrix (SCM) and apply subspace methods like MUltiple SIgnal Classification (MUSIC) [4] to find the DoAs. Because the noise subspace is required to be nontrivial, an M-element uniform linear array (ULA) can only resolve up to $M-1$ sources. To remove such a limit and reduce the cost of sensors, one can choose an N-element SLA with the same aperture but no "holes" in its co-array [5]. In this case, the M-by-M SCM of the original ULA can be reconstructed from the N-by-N SCM of the SLA. Taking a 5-element minimum redundancy array (MRA) for example, it can recover the SCM of a 10-element ULA and thus resolve up to 9 sources with only 5 sensors. Although such an exploitation on the co-array structure can deliver more degrees of freedom, an extra step of covariance matrix estimation is required [6].

The earliest approach to this problem dates back to the work by Pillai et al. in 1985, which completes a Toeplitz matrix via redundancy averaging and direct augmentation [7]. Since the SCM of a ULA is positive semidefinite and possibly Toeplitz, the matrix estimation problem can be formulated as constrained optimization problems under the well-known maximum likelihood (ML) principle. However, these problems are nontrivial due to being highly nonconvex, and one often needs to relax them into convex optimization problems. For example, the problem of the coarray ML-MUSIC (Co-MLM) [8] is usually relaxed into the SDP problem of SPA [9] according to the extended invariance principle [10], [11], with its global minimizer approximating the ML estimator as the number of snapshots approaches infinity. Besides convex relaxation, another strategy to tackle nonconvex optimization is majorization-minimization. For instance, the recently proposed StructCovMLE approach by Pote and Rao [12] majorizes the concave component by a supporting hyperplane and then solves a sequence of SDP problems to arrive at a solution. There are also many other approaches such as regularized algorithms based on nuclear norm or atomic norm minimization [13], [14], Wasserstein distance minimization [15], and proxy covariance estimation [16]. Literature on DoA estimation that primarily relies on optimization techniques is vast [17], [18], so we focus on gridless and regularizer-free approaches in this paper. For grid-based DoA estimation, we refer the reader to other references such as [19] and [20].

Received 29 August 2024; revised 14 January 2025; accepted 15 February 2025. Date of publication 21 February 2025; date of current version 12 March 2025. This work was supported in part by the IEEE Signal Processing Society Scholarship, and in part by the NSF under Grant CCF-2225617 and Grant CCF-2124929. The associate editor coordinating the review of this article and approving it for publication was Xiao Fu. *(Corresponding author: Kuan-Lin Chen.)*

The authors are with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093 USA (e-mail: kuc029@ucsd.edu; brao@ucsd.edu).

Digital Object Identifier 10.1109/TSP.2025.3544170

In the past decade, the advent of deep learning has opened up a new paradigm for DoA estimation [21], [22], [23], [24]. As the most intuitive and earliest learning-based approach, one can discretize the angle domain into a grid and then learn a classifier [22]. However, the performance of this approach is limited by the grid size and often the performance quickly saturates as the SNR increases. On the gridless side, it was not until a recent work by Wu et al. [25] that the potential of deep learning for the matrix estimation problem was shown. Based on enforcing the Toeplitz structure of the matrix, they showed that DNNs can be trained to retrieve the noiseless SCM of a ULA from the sample SCM of an MRA, and numerical results show that such an approach outperforms the SPA in most cases. However, it was reported that its performance is worse than MUSIC when the source number is small at high SNRs. Another feature that makes the approach slightly less appealing is that a separate DNN is required for each individual source number. It is unknown whether training one DNN for all source numbers can still provide good performance. In contrast to using the Toeplitz structure, the framework proposed by Barthelme and Utschick [26] enforces the structure of positive definiteness of the matrix. Although in [26] the task of interest is subarray sampling, which is different from the present paper, the method can be applied seamlessly to the matrix estimation problem here. These two approaches are probably the most relevant related work to this paper.

In this paper, we propose a new methodology that exploits the fundamental property that a subspace is invariant of the choice of the spanning basis, and answer the following question: *Is it possible for a neural network to learn the signal or noise subspace?* In particular, we formulate the DoA estimation problem as a subspace representation learning problem, and propose new empirical risk minimization problems and loss functions to train a DNN to learn subspace representations. Our approach first constructs a DNN to output a square matrix and performs eigenvalue decomposition on the Gram matrix of the square matrix to obtain unitary bases for the signal and noise subspaces, which we refer to as subspace representations. The DNN is then trained by minimizing loss functions of different dimensions based on principal angles that calculate the average degree of separation between the desired subspace and the subspace representation. In fact, with this new methodology, one can argue that learning subspaces is simpler than learning covariance matrices. Because our loss functions are invariant to the selection of bases, they create a larger solution space and thus make it easier for a DNN to learn subspace structures. Furthermore, we prove that it is possible for a neural network to approximate signal subspaces. To parallelize the computation of learning subspaces of different dimensions, we propose a new batch sampling strategy called consistent rank sampling, which greatly accelerates the training process. In addition, we propose a new gridless end-to-end approach learning DoAs directly to study the benefit of bypassing the root-MUSIC algorithm. Our methodology does not require knowledge of the sensor array positions, making it geometry-agnostic and robust to array imperfections. Under the standard assumptions of DoA estimation, numerical results show that our approach outperforms existing SDP-based and DNN-based methods across a wide range of SNRs, snapshots, and numbers of sources.

II. Preliminaries

Notations, assumptions, definitions, and the problem of interest are set up in this section. The set $\{1, 2, \cdots, n\}$ is denoted by $[n]$. The zero-mean circularly symmetric complex Gaussian distribution with covariance $\boldsymbol{\Sigma}$ is denoted by $\mathcal{CN}(\mathbf{0}, \boldsymbol{\Sigma})$. The Frobenius norm of a matrix $\mathbf{A}$ is denoted by $\|\mathbf{A}\|_F$. The trace of a matrix $\mathbf{A}$ is denoted by tr$(\mathbf{A})$. The set of n-by-n Hermitian matrices is denoted by $\mathbb{H}^n$. Given $\mathbf{A} \in \mathbb{H}^n$, $\mathbf{A} \succeq 0$ (resp., $\mathbf{A} \succ 0$) means that $\mathbf{A}$ is positive semidefinite (resp., positive definite). The set of n-by-n Toeplitz matrices is denoted by $\mathbb{T}^n$. For every $\mathbf{A} \in \mathbb{H}^n \cap \mathbb{T}^n$ whose first row is represented by a vector $\mathbf{u}$, $\mathbf{A}$ is denoted as Toep$(\mathbf{u})$. The minimum eigenvalue of a matrix $\mathbf{A} \succeq 0$ is denoted by $\lambda_{\text{min}}(\mathbf{A})$. The matrix logarithm of $\mathbf{A}$ is denoted by $\log(\mathbf{A})$ [27]. The set of all k-by-k permutation matrices is denoted by $\mathcal{P}_k$. The orthogonal projector onto a subspace $\mathcal{U}$ and the range of a matrix $\mathbf{A}$ are denoted by $P_{\mathcal{U}}$ and $P_{\mathbf{A}}$, respectively.

A. Assumptions

Let us consider an M-element ULA with spacing $d = \frac{\lambda}{2}$ centered at the origin. Assume that there are k narrowband and far-field source signals $\{s_i\}_{i=1}^k$ with a carrier wavelength λ impinging on the array from DoAs $\boldsymbol{\theta} = \{\theta_1, \theta_2, \cdots, \theta_k\} \subset [0, \pi]$. Under the plane wave assumption [5], the received array measurement vector or snapshot $\mathbf{y}(t) \in \mathbb{C}^M$ at time $t \in [T]$ can be modeled as

$$\mathbf{y}(t) = \sum_{i=1}^{k} s_i(t)\mathbf{a}(\theta_i) + \mathbf{n}(t) = \mathbf{A}(\boldsymbol{\theta})\mathbf{s}(t) + \mathbf{n}(t) \qquad (1)$$

where $\mathbf{a}(\theta) : [0, \pi] \to \mathbb{C}^M$ is the array manifold of the M-element ULA whose i-th element is given by

$$[\mathbf{a}(\theta)]_i = e^{j2\pi\left(i-1-\frac{(M-1)}{2}\right)\frac{d}{\lambda}\cos\theta}, i \in [M] \qquad (2)$$

and $\mathbf{A}(\boldsymbol{\theta}) = \begin{bmatrix}\mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) & \cdots & \mathbf{a}(\theta_k)\end{bmatrix}$. The source signal vectors are given by $\mathbf{s}(t) = \begin{bmatrix}s_1(t) & s_2(t) & \cdots & s_k(t)\end{bmatrix}^{\mathsf{T}}$ for all $t \in [T]$ and are independent and identically distributed (i.i.d.) with $\mathbf{s}(t) \sim \mathcal{CN}(\mathbf{0}, \mathbf{P})$ where $\mathbf{P} = \text{diag}(p_1, p_2, \cdots, p_k)$ and $p_i > 0$ is the power of the i-th source signal for all $t \in [T]$. The additive noises follow $\mathbf{n}(t) \sim \mathcal{CN}(\mathbf{0}, \eta\mathbf{I}_M)$ for all $t \in [T]$ which are i.i.d. and uncorrelated with $\mathbf{s}(t)$ for all $t \in [T]$. We further assume that $T \geq M$.

Let $N \leq M$ and $\mathcal{S} = \{s_1, s_2, \cdots, s_N\} \subset [M]$ such that $s_1 < s_2 < \cdots < s_N$. Then a physical N-element linear array can be created by removing the i-th sensor from a virtual M-element ULA if $i \notin \mathcal{S}$ for all $i \in [M]$. As a result, the snapshot $\mathbf{y}_{\mathcal{S}}(t) \in \mathbb{C}^N$ received on this physical N-element linear array at time $t \in [T]$ is given by $\mathbf{y}_{\mathcal{S}}(t) = \boldsymbol{\Gamma}\mathbf{y}(t)$ where $\mathbf{y}(t)$ is the snapshot received on the virtual M-element ULA and $\boldsymbol{\Gamma} \in \mathbb{R}^{N \times M}$ is a row selection matrix given by

$$[\boldsymbol{\Gamma}]_{nm} = \begin{cases} 1, & \text{if } s_n = m, \\ 0, & \text{otherwise,} \end{cases}, n \in [N], m \in [M]. \qquad (3)$$

In this paper, we are interested in $\mathcal{S}$ that gives rise to an SLA with the same aperture as the ULA and with no holes in its co-array such as an MRA [5] or a nested array [28]. The number of sources k is assumed to be given.

B. SCMs and the DoA Estimation Problem

With the above assumptions, it follows that the noiseless SCM of the ULA at every $t \in [T]$ can be written as $\mathbf{R}_0 = \mathbf{A}(\boldsymbol{\theta})\mathbf{P}\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta})$ and the noiseless SCM of the SLA is $\mathbf{R}_{\mathcal{S}} = \boldsymbol{\Gamma}\mathbf{R}_0\boldsymbol{\Gamma}^{\mathsf{T}}$. The sample SCM of the ULA and the SLA are denoted by $\hat{\mathbf{R}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{y}(t)\mathbf{y}^{\mathsf{H}}(t)$ and $\hat{\mathbf{R}}_{\mathcal{S}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{y}_{\mathcal{S}}(t)\mathbf{y}_{\mathcal{S}}^{\mathsf{H}}(t)$. Given M, k, $\mathcal{S}$, and $\hat{\mathbf{R}}_{\mathcal{S}}$, the goal of the DoA estimation problem is to recover $\boldsymbol{\theta}$. Note that it is possible that $k > N$ because $k \in [M-1]$. In this paper, we focus on gridless methods recovering an M-by-M matrix and use the root-MUSIC algorithm [29], [30] to find $\boldsymbol{\theta}$.

C. Neural Network Models

The rectified linear unit (ReLU) activation function is defined as $x \mapsto \max(0, x)$. A ReLU network can be expressed as a composition of affine functions and ReLU activation functions. We adopt the definition of ReLU networks from Definition 4 in [31]. Given a complex-valued input, we first separate it into its real and imaginary components, which are then processed by the network. The network produces corresponding real-valued outputs, which are subsequently recombined to form a complex-valued result.

III. Prior Art

According to the assumptions and settings in Section II, we will briefly review several popular or insightful approaches in the literature including the widely used SDP-based methods and recently proposed DNN-based approaches. Despite their differences, notice that most of them fall into the category of minimizing some distance between two covariance matrices in an appropriate space. The materials covered in this section will serve as important background and contrast with our main contributions detailed in Section IV.

A. The Maximum Likelihood Problem

Based on the assumptions in Section II-A, it follows that $\mathbf{R}_0 + \eta\mathbf{I}_M$ is positive semidefinite and possibly Toeplitz. Because $\mathbf{y}_{\mathcal{S}}(t) \sim \mathcal{CN}(\mathbf{0}, \mathbf{R}_{\mathcal{S}} + \eta\mathbf{I}_N)$, one can formulate the following constrained optimization problem according to the maximum likelihood principle:

$$\begin{aligned}&\min_{\mathbf{R}\in\mathbb{H}^M} && \log\det\left(\boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}}\right) + \operatorname{tr}\left(\left(\boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}}\right)^{-1}\hat{\mathbf{R}}_{\mathcal{S}}\right)\\ &\text{subject to} && \mathbf{R}\succeq 0, \quad \mathbf{R}\in\mathbb{T}^M. \end{aligned}\tag{4}$$

By minimizing the Kullback–Leibler divergence of $\mathcal{CN}(\mathbf{0}, \hat{\mathbf{R}}_{\mathcal{S}})$ from $\mathcal{CN}(\mathbf{0}, \boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}})$, one can also derive the above problem. Due to the nonconvex objective, solving (4) is nontrivial; and thus relaxing or reformulating (4) into a tractable problem is often necessary to arrive at an accepted solution. Section III-B and III-C below describe tractable optimization problems that are widely used in this context.

B. Redundancy Averaging and Direct Augmentation

Because an SLA can generate all of the autocorrelation lags of the corresponding ULA, Pillai et al. proposed the earliest approach of recovering $\mathbf{R}_0 + \eta\mathbf{I}_M$ from $\hat{\mathbf{R}}_{\mathcal{S}}$, i.e., the so-called redundancy averaging and direct augmentation approach [7]. This approach is identical to solving the following matrix augmentation problem [15]:

$$\min_{\mathbf{R}\in\mathbb{C}^{M\times M}} \quad \left\|\boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}} - \hat{\mathbf{R}}_{\mathcal{S}}\right\|_F \quad \text{subject to} \quad \mathbf{R}\in\mathbb{T}^M \tag{5}$$

which has a closed-form solution that is Hermitian and Toeplitz but not necessarily positive semidefinite. Spatial smoothing [28] can be applied to fix this issue via $\frac{1}{M}\mathbf{R}\mathbf{R}^{\mathsf{H}}$ if $\mathbf{R}$ is the solution of (5).

C. Direct SDP-Based Methods

Based on the covariance fitting criterion [32], Yang et al. formulated the SPA [9] involving the optimization problem:

$$\begin{aligned}&\min_{\mathbf{X}\in\mathbb{H}^N, \mathbf{R}\in\mathbb{H}^M} && \operatorname{tr}(\mathbf{X}) + \operatorname{tr}\left(\hat{\mathbf{R}}_{\mathcal{S}}^{-1}\boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}}\right)\\ &\text{subject to} && \begin{bmatrix}\mathbf{X} & \hat{\mathbf{R}}_{\mathcal{S}}^{\frac{1}{2}} & \\ \hat{\mathbf{R}}_{\mathcal{S}}^{\frac{1}{2}} & \boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}} & \\ & & \mathbf{R}\end{bmatrix}\succeq 0, \quad \mathbf{R}\in\mathbb{T}^M. \end{aligned}\tag{6}$$

The noiseless SCM is then estimated by $\mathbf{R} - \lambda_{\min}(\mathbf{R})\mathbf{I}_M$ where $\mathbf{R}$ is the solution of (6). Another interesting approach based on the Bures-Wasserstein distance [33] was developed by Wang et al. [15]. The optimization problem is given by

$$\begin{aligned}&\min_{\mathbf{X}\in\mathbb{C}^{N\times N}, \mathbf{R}\in\mathbb{H}^M} && \operatorname{tr}\left(\hat{\mathbf{R}}_{\mathcal{S}} + \boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}} - \mathbf{X} - \mathbf{X}^{\mathsf{H}}\right)\\ &\text{subject to} && \begin{bmatrix}\boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}} & \mathbf{X}\\ \mathbf{X}^{\mathsf{H}} & \hat{\mathbf{R}}_{\mathcal{S}}\end{bmatrix}\succeq 0, \quad \mathbf{R}\succeq 0, \quad \mathbf{R}\in\mathbb{T}^M. \end{aligned}\tag{7}$$

Both optimization problems in (6) and (7) are SDPs that can be solved by off-the-shelf solvers such as the SDPT3 [34].

D. Majorization-Minimization

Since the term $\log\det(\cdot)$ in (4) is concave on the positive semidefinite cone and the trace term can be written as an SDP via the Schur complement lemma, majorization-minimization algorithms can be used to tackle (4). Using a supporting hyperplane to majorize the term $\log\det$, one can derive the so-called "StructCovMLE" approach [12]. Let $\mathbf{R}^{(0)}$ be initialized to $\mathbf{I}_M$. For $i = 0, 1, 2, \cdots$, StructCovMLE calculates the iterate $\mathbf{R}^{(i+1)}$ by solving the optimal $\mathbf{R}$ in the following SDP:

$$\begin{aligned}&\min_{\mathbf{R}\in\mathbb{H}^M, \mathbf{X}\in\mathbb{H}^N} && \operatorname{tr}\left(\left(\boldsymbol{\Gamma}\mathbf{R}^{(i)}\boldsymbol{\Gamma}^{\mathsf{T}}\right)^{-1}\boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}}\right) + \operatorname{tr}\left(\mathbf{X}\hat{\mathbf{R}}_{\mathcal{S}}\right)\\ &\text{subject to} && \begin{bmatrix}\mathbf{X} & \mathbf{I}_N & \\ \mathbf{I}_N & \boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}} & \\ & & \mathbf{R}\end{bmatrix}\succeq 0, \quad \mathbf{R}\in\mathbb{T}^M. \end{aligned}\tag{8}$$

The final solution is then obtained through running a number of iterations until a stopping criterion is satisfied. For example, the relative change between $\mathbf{R}^{(i)}$ and $\mathbf{R}^{(i+1)}$ being sufficiently small. As there is a sequence of SDPs to be solved, the complexity of this approach is greater than the complexity of the above direct SDP-based methods in Section III-C.

E. Proxy Covariance Matrix Estimation

Instead of estimating the covariance matrix, Sarangi et al. [16] proposed a "proxy covariance matrix" approach (Prox-Cov) that jointly calculates a positive definite weighting matrix $\mathbf{W}$ and a proxy covariance $\mathbf{R}$ such that the weighted covariance matrix from the data best fits the proxy covariance. Based on this rationale, they formulated the following SDP:

$$\begin{aligned} &\min_{\mathbf{R}\in\mathbb{H}^M,\mathbf{W}\in\mathbb{H}^T} \quad \left\| \mathbf{Y}\mathbf{W}\mathbf{Y}^{\mathsf{H}} - \boldsymbol{\Gamma}\mathbf{R}\boldsymbol{\Gamma}^{\mathsf{T}} \right\|_F^2 \\ &\text{subject to} \quad \mathbf{R} \succeq 0, \quad \mathbf{R} \in \mathbb{T}^M, \quad \mathbf{W} \succeq \epsilon \mathbf{I}_T \end{aligned} \tag{9}$$

where $\mathbf{Y} = \begin{bmatrix} \mathbf{y}(1) & \mathbf{y}(2) & \cdots & \mathbf{y}(T) \end{bmatrix}$ is a matrix whose columns are all of the received snapshots and ϵ is a hyperparameter which is strictly positive. An interesting property of (9) is that it can exactly recover the signal subspace, overcoming the shortcoming of (5), under appropriate assumptions [16]. Unlike the aforementioned methods that estimate a covariance matrix from a sample SCM, Prox-Cov considers all snapshots and attempts to estimate the signal and noise subspaces by introducing a weighting matrix $\mathbf{W}$, which allows for arbitrary signal powers while maintaining the same range space from the snapshots.

F. DNN-Based Covariance Matrix Reconstruction

Let $\mathcal{D} = \left\{ \hat{\mathbf{R}}_{\mathcal{S}}^{(l)}, \mathbf{R}_0^{(l)} \right\}_{l=1}^{L}$ be a dataset containing L pairs of matrices where every $\hat{\mathbf{R}}_{\mathcal{S}}^{(l)} \in \mathbb{H}^N$ is a sample SCM of the N-element SLA and $\mathbf{R}_0^{(l)} \in \mathbb{H}^M$ is the corresponding noiseless SCM of the M-element ULA. According to the work by Barthelme and Utschick [26], one can formulate the matrix estimation problem as a learning problem whose goal is to find optimal parameters W^* of a DNN model $f_W : \mathbb{C}^{N\times N} \to \mathbb{C}^{M\times M}$ such that $f_{W^*}(\hat{\mathbf{R}}_{\mathcal{S}}) f_{W^*}^{\mathsf{H}}(\hat{\mathbf{R}}_{\mathcal{S}}) \approx \mathbf{R}_0$ for every possible pair $(\hat{\mathbf{R}}_{\mathcal{S}}, \mathbf{R}_0)$ of interest. The Gram matrix here is to ensure the positive semidefiniteness. The search of W^* is done through the training of the DNN. After training, the function f_{W^*} is evaluated at an N-by-N sample SCM to obtain an M-by-M SCM estimate. The model f_W is trained by solving the empirical risk minimization problem

$$\min_{W} \quad \frac{1}{L} \sum_{l=1}^{L} d\left(f_W\left(\hat{\mathbf{R}}_{\mathcal{S}}^{(l)}\right) f_W^{\mathsf{H}}\left(\hat{\mathbf{R}}_{\mathcal{S}}^{(l)}\right), \mathbf{R}_0^{(l)} \right) \tag{10}$$

where d is a metric or distance. For example, the well-known Frobenius norm

$$d_{\text{Fro}}(\mathbf{E}, \mathbf{F}) = \|\mathbf{E} - \mathbf{F}\|_F \tag{11}$$

and the *affine invariant distance* [35]

$$d_{\text{Aff}}(\mathbf{E}, \mathbf{F}) = \left\| \log\left(\mathbf{F}^{-\frac{1}{2}} \mathbf{E} \mathbf{F}^{-\frac{1}{2}} \right) \right\|_F \tag{12}$$

that gives the length of the shortest curve between the two points in the convex cone of all positive definite matrices $\{\mathbf{E} \in \mathbb{H}^M \mid \mathbf{E} \succ 0\}$. If (12) is used in (10), $\mathbf{R}_0^{(l)}$ is replaced by $\mathbf{R}_0^{(l)} + \delta \mathbf{I}_M$ for some $\delta > 0$ as $\mathbf{R}_0^{(l)}$ can be singular. Although this method by Barthelme and Utschick [26] was originally developed for the subarray sampling problem, we find that it is suitable for the matrix estimation problem in this paper.

An early study in the literature addressing the matrix estimation problem using a DNN is the work of Wu et al. [25]. Let $\mathbf{u} \in \mathbb{C}^M$ be the vector representing the first row of $\mathbf{A}(\boldsymbol{\theta})\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta})$. Instead of using the Gram matrix to generate a positive semidefinite matrix output, Wu et al. constructed a DNN $f_{W_k} : \mathbb{C}^{N\times N} \to \mathbb{C}^M$ to estimate $\mathbf{u}$ and then recovered the matrix by $\text{Toep}\left(f_{W_k}(\hat{\mathbf{R}}_{\mathcal{S}})\right)$ for a given $\hat{\mathbf{R}}_{\mathcal{S}}$ and source number k. The models $\{f_{W_k}\}_{k=1}^{M-1}$ were trained individually by the squared loss function

$$d_{\text{squ}}(\mathbf{u}, \mathbf{v}) = \frac{1}{2M} \|\mathbf{u} - \mathbf{v}\|_2^2 . \tag{13}$$

Though $M-1$ DNNs are used in [25], note that this method is not limited by the number of DNNs used. The Toeplitz prior and d_{squ} can be used to train a single network if desired.

IV. Subspace Representation Learning

A weakness of the above DNN-based methods is that their loss functions are not invariant to a different matrix representation of the signal or noise subspace. To elaborate, let $\boldsymbol{\Sigma} \in \mathbb{H}^K$ be any positive definite matrix such that $\boldsymbol{\Sigma} \neq \mathbf{P}$. Then, $\mathbf{A}(\boldsymbol{\theta})\boldsymbol{\Sigma}\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta})$ and $\mathbf{R}_0$ have exactly the same signal subspace $\left\{ \mathbf{A}(\boldsymbol{\theta})\mathbf{x} \middle| \mathbf{x} \in \mathbb{C}^K \right\}$ that leads to the same DoAs via the root-MUSIC algorithm. However, $\mathbf{A}(\boldsymbol{\theta})\boldsymbol{\Sigma}\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta}) \neq \mathbf{R}_0$ which implies $d\left(\mathbf{A}(\boldsymbol{\theta})\boldsymbol{\Sigma}\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta}), \mathbf{R}_0\right) > 0$ for any metric or distance d on $\mathbb{C}^{M\times M}$. If $\boldsymbol{\Sigma} = \rho \mathbf{I}_K$, it can be easily seen that $d \to \infty$ as $\rho \to \infty$ for most of the common distances such as d_{Fro} and d_{squ} mentioned above even though the signal subspace induced by $\mathbf{A}(\boldsymbol{\theta})\boldsymbol{\Sigma}\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta})$ is always the same as the one induced by $\mathbf{R}_0$. This is not a desirable property for a loss function because it significantly reduces the solution space and makes it much more difficult to find and approximate the signal or noise subspace. It is worth noting that many existing methods (e.g., most of the methods in Section III) measure the goodness of fit via some distance between two covariance matrices, effectively solving a harder problem than needed. Because the root-MUSIC algorithm only requires the knowledge of the signal or noise subspace, the problem of covariance estimation is actually harder than DoA estimation.

To address the above-mentioned issue, we propose a new methodology which we call *subspace representation learning*. In the subsections below, we will first introduce a new output representation for DNN models to establish the invariance to the choice of $\boldsymbol{\Sigma}$. Next, we construct a novel family of loss functions to train these DNN models based on the goodness of subspace fitting and show that it is possible for a DNN to approximate signal subspaces. The root-MUSIC algorithm is then applied on the learned signal subspace to obtain the DoAs. We then discuss the use case for imperfect arrays. Finally, we propose a new batch sampling approach to parallelize the computation involved during training.

A. Subspace Representations of Different Dimensions

Because every k-dimensional subspace $\mathcal{U}_k$ of $\mathbb{C}^M$ is a point in the *Grassmann manifold* or *Grassmannian* $\text{Gr}(k, M)$, we construct a DNN model f_W such that

$$f_W : \mathbb{C}^{N\times N} \times [M-1] \to \bigcup_{k=1}^{M-1} \mathrm{Gr}(k,M). \tag{14}$$

The codomain is a union of $M-1$ Grassmannians. To represent points of this union numerically, we can pick any matrix $\mathbf{U} \in \mathbb{C}^{M\times k}$ whose columns represent a unitary basis of $\mathcal{U}_k \in \mathrm{Gr}(k,M)$ for all $k \in [M-1]$. Based on this perspective, the model f_W is instructed to generate a matrix $\mathbf{X} \in \mathbb{C}^{M\times M}$ whose Gram matrix is factorized by eigenvalue decomposition:

$$\mathbf{X}\mathbf{X}^{\mathsf{H}} = \begin{bmatrix} \tilde{\mathbf{U}} & \tilde{\mathbf{V}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_k & \\ & \boldsymbol{\Lambda}_{M-k} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{U}}^{\mathsf{H}} \\ \tilde{\mathbf{V}}^{\mathsf{H}} \end{bmatrix} \tag{15}$$

where $\boldsymbol{\Lambda}_k$ and $\boldsymbol{\Lambda}_{M-k}$ are diagonal matrices representing the k largest eigenvalues and $M-k$ smallest eigenvalues, respectively; and the columns of $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ are their corresponding orthonormal eigenvectors, respectively. Since the columns of $\tilde{\mathbf{U}} \in \mathbb{C}^{M\times k}$ form a unitary basis, a subspace $\tilde{\mathcal{U}}_k \in \mathrm{Gr}(k,M)$ can then be identified by the range space of $\tilde{\mathbf{U}}$ and thus the function f_W can generate points in the union of the Grassmannians. As long as $\mathbf{X}$ maintains the same signal subspace, the subspace $\tilde{\mathcal{U}}$ generated by f_W is invariant to the change of $\mathbf{X}$. One simple invariance can be easily seen by changing the eigenvalues while maintaining the order of $\boldsymbol{\Lambda}_k$ and $\boldsymbol{\Lambda}_{M-k}$. The subspace $\tilde{\mathcal{U}}$ is also invariant to the equivalence class of its unitary bases.

Given a dataset $\mathcal{D} = \{\hat{\mathbf{R}}_{\mathcal{S}}^{(l)}, \mathbf{R}_0^{(l)}\}_{l=1}^{L}$, we extract the signal subspace $\mathcal{U}^{(l)}$ of $\mathbf{R}_0^{(l)}$ for every $l \in [L]$ via eigenvalue decomposition to create target subspace representations. Note that $\mathcal{U}^{(l)}$ can also be identified from $\mathbf{A}(\boldsymbol{\theta}^{(l)})\mathbf{A}^{\mathsf{H}}(\boldsymbol{\theta}^{(l)})$ if only a dataset of $\{\hat{\mathbf{R}}_{\mathcal{S}}^{(l)}, \boldsymbol{\theta}^{(l)}\}$ is available.

B. Distances Between Subspace Representations

To learn the target subspace representations in $\mathcal{D}$, we find the parameters W by solving the following empirical risk minimization problem

$$\min_{W} \quad \frac{1}{L}\sum_{l=1}^{L} d_{k=k^{(l)}}\left(f_W\left(\hat{\mathbf{R}}_{\mathcal{S}}^{(l)}, k^{(l)}\right), \mathcal{U}^{(l)}\right) \tag{16}$$

where $d_k : \mathrm{Gr}(k,M) \times \mathrm{Gr}(k,M) \to [0,\infty)$ is some distance on the Grassmannian $\mathrm{Gr}(k,M)$. We propose to construct d_k as a function of the vector of *principal angles* between two given subspaces because it is a necessary condition if d_k is invariant to any rotation in the unitary group $\mathbb{U}(M)$ of M-by-M unitary matrices [36], i.e.,

$$d_k(\mathbf{Q}\cdot\mathcal{U}, \mathbf{Q}\cdot\tilde{\mathcal{U}}) = d_k(\mathcal{U},\tilde{\mathcal{U}}) \tag{17}$$

for every $\mathcal{U},\tilde{\mathcal{U}} \in \mathrm{Gr}(k,M)$ and every $\mathbf{Q} \in \mathbb{U}(M)$. The *left action* of $\mathbb{U}(M)$ on $\mathrm{Gr}(k,M)$ in (17) is defined by $\mathbf{Q}\cdot\mathcal{U} := \mathrm{span}\,(\mathbf{QB})$ where the columns of $\mathbf{B} \in \mathbb{C}^{M\times k}$ form a basis of $\mathcal{U}$. According to Theorem 1 of [37], the principal angles $\boldsymbol{\phi}_k = \begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_k \end{bmatrix}^{\mathsf{T}}$ between $\mathcal{U} \in \mathrm{Gr}(k,M)$ and $\tilde{\mathcal{U}} \in \mathrm{Gr}(k,M)$ can be calculated by

$$\phi_i(\mathcal{U},\tilde{\mathcal{U}}) = \cos^{-1}\left(\sigma_i(\mathbf{U}^{\mathsf{H}}\tilde{\mathbf{U}})\right) \tag{18}$$

for $i \in [k]$ where $\mathbf{U} \in \mathbb{C}^{M\times k}$ and $\tilde{\mathbf{U}} \in \mathbb{C}^{M\times k}$ are matrices whose columns form unitary bases of $\mathcal{U}$ and $\tilde{\mathcal{U}}$, respectively, and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$ are the singular values of the singular value decomposition of $\mathbf{U}^{\mathsf{H}}\tilde{\mathbf{U}}$. As $\cos^{-1}$ is a monotonically decreasing function over its domain, the principal angles satisfy $\phi_1 \leq \phi_2 \leq \cdots \leq \phi_k$.

TABLE I
DISTANCES BETWEEN SUBSPACES

Distance	Function of Principal Angles
Geodesic (arc length)	$\lVert\boldsymbol{\phi}_k\rVert_2$
Fubini-Study	$\cos^{-1}\left(\prod_{i=1}^{k}\cos\phi_i\right)$
Chordal (projection Frobenius norm)	$\left(\sum_{i=1}^{k}\sin^2\phi_i\right)^{\frac{1}{2}}$
Projection 2-norm	$\sin\phi_k$
Chordal Frobenius norm	$2\left(\sum_{i=1}^{k}\sin^2\frac{\phi_i}{2}\right)^{\frac{1}{2}}$
Chordal 2-norm	$2\sin\frac{\phi_k}{2}$

Several examples of distances based on $\boldsymbol{\phi}_k$ [38], [39], [40], [41] are provided in Table I. Among them, the most natural choice of d_k is the *geodesic distance* [36]

$$d_k^{\mathrm{Geo}}(\mathcal{U},\tilde{\mathcal{U}}) = \left\lVert\boldsymbol{\phi}_k\left(\mathcal{U},\tilde{\mathcal{U}}\right)\right\rVert_2 = \left(\sum_{i=1}^{k}\phi_i^2\left(\mathcal{U},\tilde{\mathcal{U}}\right)\right)^{\frac{1}{2}} \tag{19}$$

which defines the length of the shortest curve between the two points $\mathcal{U}$ and $\tilde{\mathcal{U}}$ on the Grassmannian $\mathrm{Gr}(k,M)$. The geodesic distance of any two points on $\mathrm{Gr}(k,M)$ is bounded from above by $\sqrt{k}\frac{\pi}{2}$ [36]; and one can easily construct different loss functions which are bounded.

C. Approximation

In this subsection, we attempt to enhance the feasibility of subspace representation learning from an approximation viewpoint. In particular, we present a guarantee for a neural network model to approximate the signal subspace.

Theorem 1: For every $k \in [M-1]$ and every $\epsilon > 0$, there exists a ReLU network $f : \mathbb{C}^{N\times N} \to \mathrm{Gr}(k,M)$ such that

$$\int_{[0,\pi]^k} d_k^{\mathrm{Geo}}\left(f\left(\mathbf{R}_{\mathcal{S}}\right), P_{\mathbf{A}(\boldsymbol{\theta})}\right) d\boldsymbol{\theta} < \epsilon. \tag{20}$$

The proof of Theorem 1 is contained in Appendix A. Here, subspaces are represented by their orthogonal projectors to ensure every $\mathcal{U} \in \mathrm{Gr}(k,M)$ has a unique representation. In other words, $\mathrm{Gr}(k,M)$ is equivalent to

$$\left\{P \in \mathbb{C}^{M\times M} \mid P^{\mathsf{H}} = P, P^2 = P,\ \mathrm{rank}(P) = k\right\}. \tag{21}$$

If the ideal covariance matrices are used, Theorem 1 shows that the average geodesic distance between the predicted subspaces and the desirable signal subspaces can be made arbitrarily small when a suitable ReLU network is picked. From an array processing point of view, it is trivial that the signal subspace can always be extracted from $\mathbf{R}_{\mathcal{S}}$. However, Theorem 1 illustrates that this process can be achieved up to a small error by evaluating a continuous piecewise linear function [31]. In order to sketch the proof, notice that a simple distance on $\text{Gr}(k,M)$ can be constructed by

$$(\mathcal{U}_1,\mathcal{U}_2)\mapsto \|P_{\mathcal{U}_1}-P_{\mathcal{U}_2}\|_F\,. \tag{22}$$

Lemma 1 below shows that the geodesic distance can be bounded from above by the composition of a strictly increasing function and the simple distance in (22), allowing us to leverage the continuity of the orthogonal projection operator in an appropriate manner to prove Theorem 1. It may be possible to extend Theorem 1 to a more realistic case using $\hat{\mathbf{R}}_{\mathcal{S}}$ with a probabilistic guarantee.

Lemma 1: For every $\mathcal{U}_1,\mathcal{U}_2\in\text{Gr}(k,M)$ where $k\in[M-1]$,

$$d_k^{\text{Geo}}(\mathcal{U}_1,\mathcal{U}_2)\le\sqrt{k}\sin^{-1}\left(\frac{\|P_{\mathcal{U}_1}-P_{\mathcal{U}_2}\|_F}{\sqrt{2}}\right). \tag{23}$$

The proof of Lemma 1 is contained in Appendix B. Lemma 1 ensures that $\|P_{\mathcal{U}_1}-P_{\mathcal{U}_2}\|_F\to 0$ implies $d_k^{\text{Geo}}(\mathcal{U}_1,\mathcal{U}_2)\to 0$.

D. Learning With Imperfect Arrays

Sensor arrays are not perfect in reality. For example, the array manifold may be corrupted by several imperfections including the gain bias, phase bias, sensor position error, and the intersensor mutual coupling [21]. Because model-based methods such as SDP-based approaches in (6) and (7) often rely on prior knowledge of the sensor positions $\mathcal{S}$ to create $\boldsymbol{\Gamma}$ in their optimization problems, they are not robust to sensor position errors; and fixing such a model mismatch is nontrivial. In contrast, our methodology does not suffer from this model mismatch issue due to its geometry-agnostic or imperfection-agnostic nature. The empirical risk minimization problem we solve in the imperfect array case is still (16). As described in the last paragraph of Section IV-A, $\mathcal{U}^{(l)}$ can be identified from the ground truth $\boldsymbol{\theta}^{(l)}$; and $\hat{\mathbf{R}}_{\mathcal{S}}$ is the sample SCM from the imperfect array. Hence, both the problem formulation in (16) and the model (14) do not depend on the sensor positions. In addition, our method does not need to know the array is imperfect and the degree of imperfections. The information is already embedded in the dataset and solving (16) will enforce the DNN model to learn the subspace representations of a perfect virtual ULA from the imperfect array.

E. Consistent Rank Sampling

To learn subspaces of different dimensions in one DNN model, the empirical risk minimization problem (16) requires $M-1$ loss functions $d_1,d_2,\cdots,d_{M-1}$ that calculate unitary bases of different dimensions from 1 to $M-1$. Although (16) can be solved by the well-known minibatch stochastic gradient descent (SGD) algorithm, it is hard for the computation of different dimensions to be parallelized on a graphics processing unit (GPU). To fix this issue, we propose *consistent rank sampling*, a new batch sampling strategy for learning subspaces of different dimensions in one DNN model. Instead of randomly sampling from $\mathcal{D}$, we propose randomly sampling a batch of data points whose source number k is consistent from $\mathcal{D}$. This way, only one d_k needs to be evaluated in every gradient step, streamlining the computation of unitary bases in the same dimension k. It is important to note that consistent rank sampling is a crucial strategy to make training efficient. Without this strategy, training DNNs on large datasets becomes extremely difficult due to the slow training speed. Although the strategy is developed for subspace representation learning, it is generally applicable to empirical risk minimization problems that involve loss functions of different dimensions.

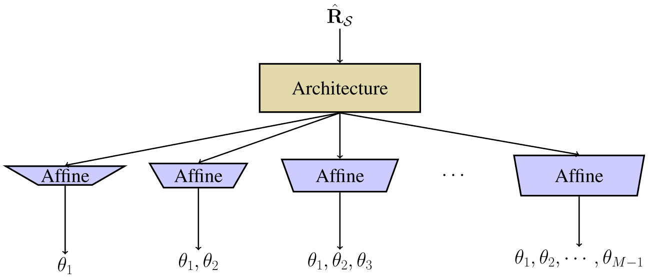

$\hat{\mathbf{R}}_{\mathcal{S}}$
Architecture
Affine
Affine
Affine
$\cdots$
Affine
θ_1
θ_1,θ_2
$\theta_1,\theta_2,\theta_3$
$\theta_1,\theta_2,\cdots,\theta_{M-1}$

Fig. 1. An illustration of the gridless end-to-end model, which consists of an architecture and several output layers. The model simultaneously generates DoAs for every possible number of sources so there are $M-1$ heads (affine functions) at the output. The k-th head is picked when there are k sources.

V. A Gridless End-to-End Approach

The subspace representation learning approach utilizes the root-MUSIC algorithm on the obtained subspaces to estimate the DoAs. A natural question to study here is the following: *Is it possible to bypass the root-MUSIC algorithm and directly learn a model to output the DoAs in a gridless manner?* The best-known end-to-end approach is probably the work [22] by Papageorgiou et al.; however, it relies on a grid. The approach of mean cyclic error (MCE) network or MCENet [23] by Barthelme and Utschick is a gridless end-to-end approach but it was designed for subarray sampling and not for more sources than sensors. Below, we propose a new gridless end-to-end approach that is tailored to the localization of more sources than sensors using an SLA.

As illustrated in Fig. 1, we propose to construct a DNN model g_W such that

$$g_W:\mathbb{C}^{N\times N}\times[M-1]\to\mathbb{R}^1\times\mathbb{R}^2\times\cdots\mathbb{R}^{M-1}. \tag{24}$$

The codomain is the $(M-1)$-ary Cartesian product of Euclidean spaces $\mathbb{R}^1,\mathbb{R}^2,\cdots,\mathbb{R}^{M-1}$. These Euclidean spaces are viewed as different "heads" at the output of the model where the k-dimensional Euclidean space represents the k-th head. The k-th head will be picked when there are k sources such that an element from $\mathbb{R}^k$ can represent k angles. Let $r(i)=\frac{i(i-1)}{2}$ for $i=1,2,\cdots,M$ and denote $h_k:\mathbb{R}^{r(M)}\to\mathbb{R}^k$ the projection

$$\left(x_1,\cdots,x_{r(M)}\right)\mapsto\left(x_{r(k)+1},x_{r(k)+2},\cdots,x_{r(k)+k}\right). \tag{25}$$

The empirical risk minimization problem of the gridless end-to-end model g_W is then formulated as follows

$$\min_{W}\quad\frac{1}{L}\sum_{l=1}^{L}d_{k=k^{(l)}}\left(h_{k=k^{(l)}}\circ g_W\left(\hat{\mathbf{R}}_{\mathcal{S}}^{(l)},k^{(l)}\right),\boldsymbol{\theta}^{(l)}\right) \tag{26}$$

where $d_1, d_2, \cdots, d_{M-1}$ are loss functions of different dimensions that calculate some minimum distances among all permutations. Taking the squared loss for example,

$$d_k\left(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}\right) = \frac{1}{k} \min_{\boldsymbol{\Pi} \in \mathcal{P}_k} \left\| \boldsymbol{\Pi} \hat{\boldsymbol{\theta}} - \boldsymbol{\theta} \right\|_2^2 \tag{27}$$

for $k = 1, 2, \cdots, M-1$. The minimum in (27) is equivalent to the squared loss applied to the corresponding sorted arguments according to the rearrangement inequality [42]. Because a loss function of different dimensions is adopted, the consistent rank sampling strategy detailed in Section IV-E can be applied to accelerate training on GPUs.

VI. Numerical Results

In this section, we will compare our new methodologies with existing methods including the SPA [9], the Wasserstein distance based approach (WDA) [15], the DNN-based covariance reconstruction (DCR) approach based on the Toeplitz prior [25] (DCR-T), and the DCR approach based on the Gram matrix [26] (DCR-G). In particular, we use both the Frobenius norm and the affine invariant distance for DCR-G, leading to two methods termed DCR-G-Fro and DCR-G-Aff. We do not include StructCovMLE [12] and Prox-Cov [16] because the performance of StructCovMLE was similar to SPA, and Prox-Cov [16] did not yield better performance than the SPA in our preliminary experiments. In subspace representation learning, the geodesic distance in (19) is picked for d_k, if not explicitly specified.

Below, we will first set up the scenarios for the DoA estimation problem. Next, we will describe the DNN architectures and the training procedures for the DNN-based approaches. Finally, for a given SNR and number of snapshots T, we will compare performance of different approaches in terms of the mean squared error (MSE)

$$\frac{1}{L_{\text{test}}} \sum_{l=1}^{L_{\text{test}}} \frac{1}{k} \min_{\boldsymbol{\Pi} \in \mathcal{P}_k} \left\| \boldsymbol{\Pi} \hat{\boldsymbol{\theta}}_l - \boldsymbol{\theta}_l \right\|_2^2 \tag{28}$$

for different source numbers $k \in [M-1]$ where L_{test} is the total number of random trials, $\boldsymbol{\theta}_l$ is the vector of DoAs of the ground truth at the l-th trial, and $\hat{\boldsymbol{\theta}}_l$ is the corresponding estimate given by a method of interest.

A. Settings

The physical array is an N-element MRA with $N = 5$ and $\mathcal{S} = \{1, 2, 5, 8, 10\}$, leading to a 10-element virtual ULA or $M = 10$. A study for different MRAs is deferred to Section VI-B3. Below we describe the test or evaluation conditions. The number of snapshots T is set to 50, if not explicitly specified. The SNR is defined as $10 \log_{10}\left(\frac{\frac{1}{k}\sum_{i=1}^{k} p_i}{\eta}\right)$ and we assume equal source powers $p_1 = p_2 = \cdots = p_k$, if not explicitly specified. The SNR is set to 20 dB if not explicitly stated. The finite set of SNRs $\{-10, -8, -6, \cdots, 16, 18, 20\}$ is picked when a range of SNRs is required for evaluation. The number of sources k can be any number in the set $[M-1]$. For any $k \in [M-1]$, the DoAs $\theta_1, \theta_2, \cdots, \theta_k$ are uniformly selected at random in the range $\left[\frac{1}{6}\pi, \frac{5}{6}\pi\right]$ with a minimum separation constraint $\min_{i \neq j} |\theta_i - \theta_j| \geq \frac{1}{45}\pi$. For every given SNR, T, and $k \in [M-1]$, there are 100 trials of random source signals and noises for a given $\boldsymbol{\theta}$, and there are in total 100 random $\boldsymbol{\theta}$, leading to a total number of trials $L_{\text{test}} = 10^4$ for each case. All SDP problems are solved by the SDPT3 [34] solver in CVX [43], [44].

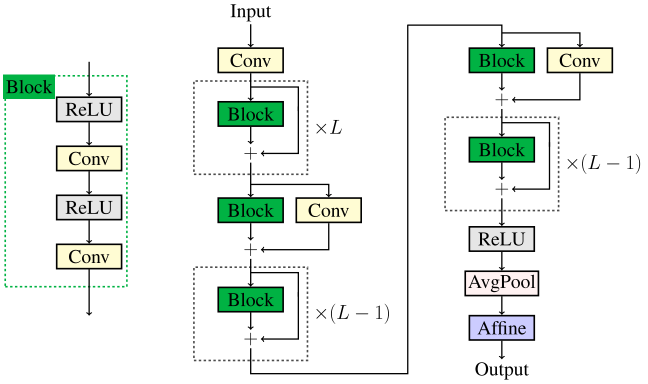
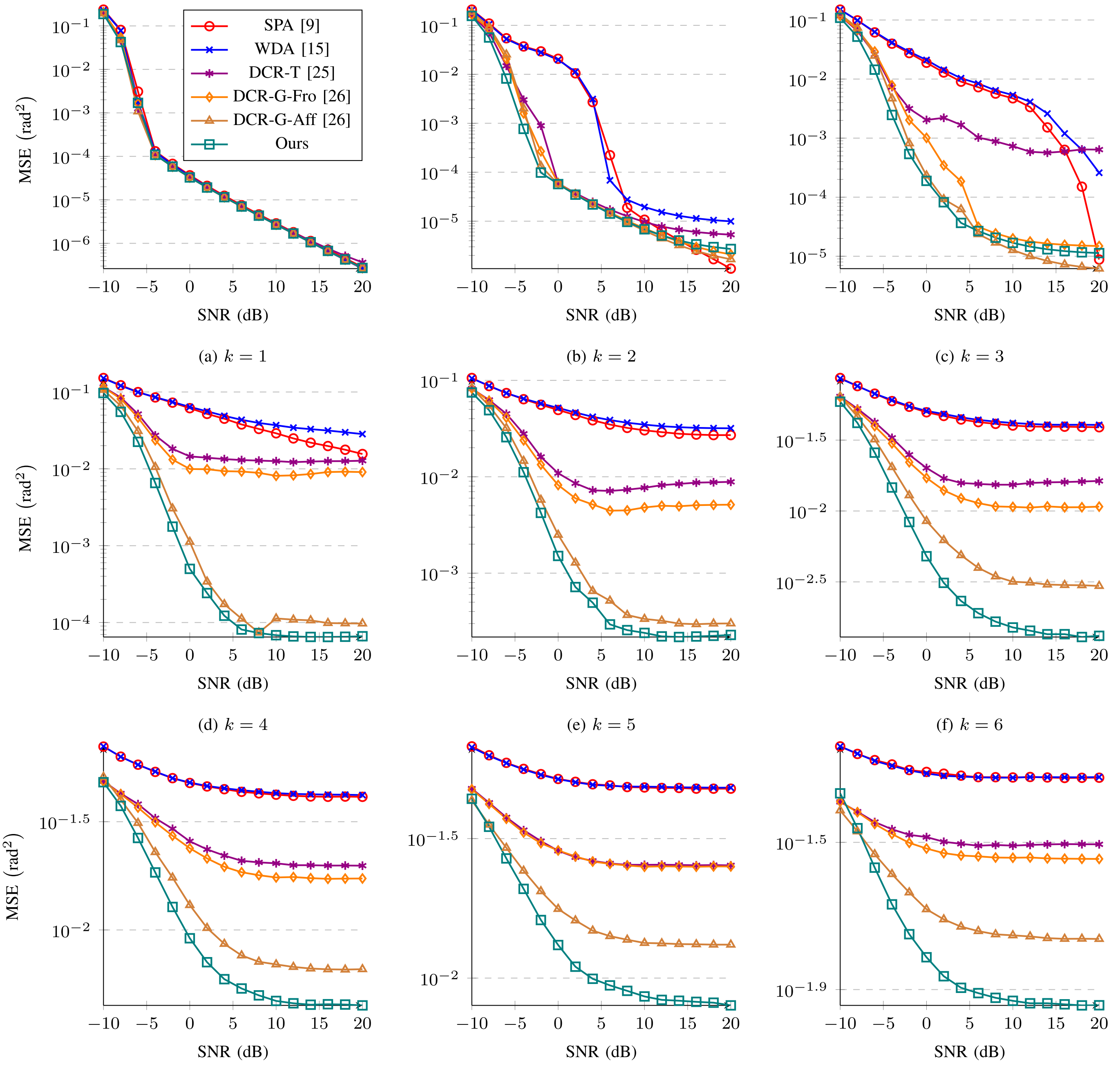


Fig. 2. An illustration of a 3-stage L-block ResNet model [45]. In the wide ResNet 16-8 (WRN-16-8) [46], there are $L = 2$ blocks per stage, leading to 16 layers in total. The widening factor is 8, meaning that WRN-16-8 is 8 times wider than the original ResNet. See Section VI-A1 for more details.

1) DNN Models: As illustrated in Fig. 2, we use WRN-16-8 [46] without the batch normalization. The pair of numbers 16-8 implies that the total number of layers is 16 and the widening factor is 8. The ReLU activation function is adopted by all of the nonlinearities in the network. All of the residual blocks are in the pre-activation form [47]. Note that wide ResNets avoid the degradation problem and enjoy certain optimization guarantees under mild assumptions [48]. The network takes an input tensor in $\mathbb{R}^{2 \times N \times N}$ and generates an output tensor in $\mathbb{R}^{2 \times M \times M}$ ($\mathbb{R}^{2 \times M}$ for DCR-T). Given an N-by-N complex matrix, it is represented by its real and imaginary parts as inputs to the network. The first and second planes of the output tensor represent the real and imaginary parts of an M-by-M complex matrix, respectively. The number of parameters is approximately 11 million. All DNN-based methods use the same architecture. The output layer is an affine function whose output dimension is tailored to each approach.

2) Training: The minibatch SGD algorithm with Nesterov momentum is used to train all of the DNN models. The momentum is set to 0.5 and the batch size is 4096. The weight decay is set to 0. All of the models are trained for 50 epochs with the one-cycle learning rate scheduler [49]. The best maximum learning rate of the scheduler for each approach is found through a grid search whose description is deferred to Appendix C. The learning rates for DCR-T, DCR-G-Fro, DCR-G-Aff, and our approach are 0.05, 0.01, 0.005, and 0.1, respectively. The weights in all models are initialized using normal distributions [50]. The value of δ is set to 10^{-4} in the DCR-G-Aff approach. For each $k \in [M-1]$, there are 2×10^6 and 6×10^5 random data points for training and validation, respectively, leading to a training dataset of size $L_{\text{train}} = 9 \times 2 \times 10^6$ and a validation dataset of size $L_{\text{val}} = 9 \times 6 \times 10^5$. For each



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$

(d) $k = 4$ (e) $k = 5$ (f) $k = 6$

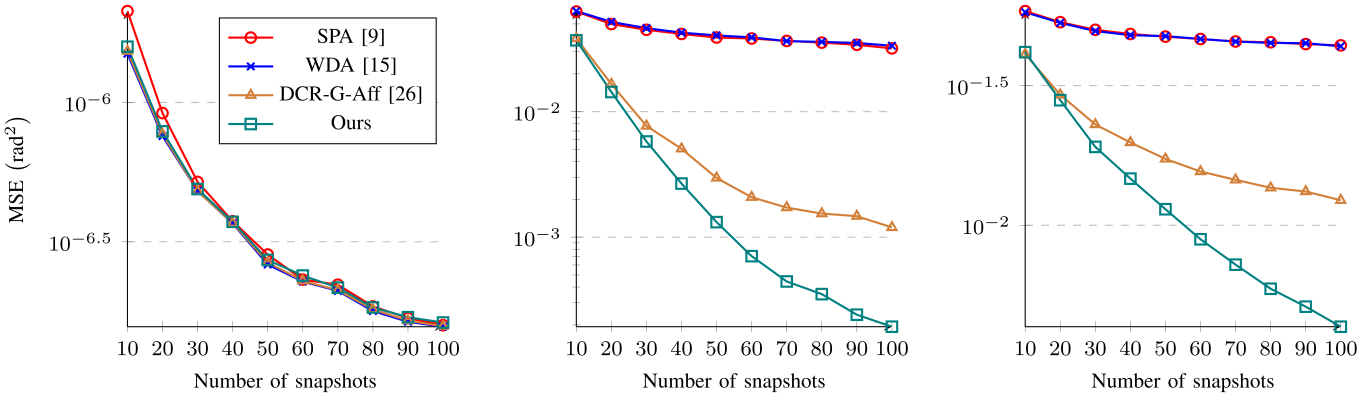
(g) $k = 7$ (h) $k = 8$ (i) $k = 9$

Fig. 3. MSE vs. SNR. Our approach is in general superior to all of the baselines. In most cases, it is significantly better than SPA, WDA, DCR-T, and DCR-G-Fro. DCR-G-Aff is the most competitive baseline. For $k > 3$, our approach outperforms DCR-G-Aff. In comparison to DCR-G-Aff at $k = 2$ or $k = 3$, our approach is slightly better at low SNRs but worse at high SNRs.

data point, the source signals and noises are generated randomly according to the assumptions in Section II-A. The SNR in decibels is uniformly picked at random in the finite set $\{-11, -9, -7, \cdots, 17, 19, 21\}$. The DoAs in the vector $\boldsymbol{\theta}$ are uniformly selected at random in the range $\left[\frac{1}{6}\pi, \frac{5}{6}\pi\right]$ with a minimum separation constraint $\min_{i \neq j} |\theta_i - \theta_j| \geq \frac{1}{60}\pi$. The sources are assumed to have equal power, if not explicitly specified. The number of snapshots is set to 50. PyTorch is used to train all the DNN models [51].

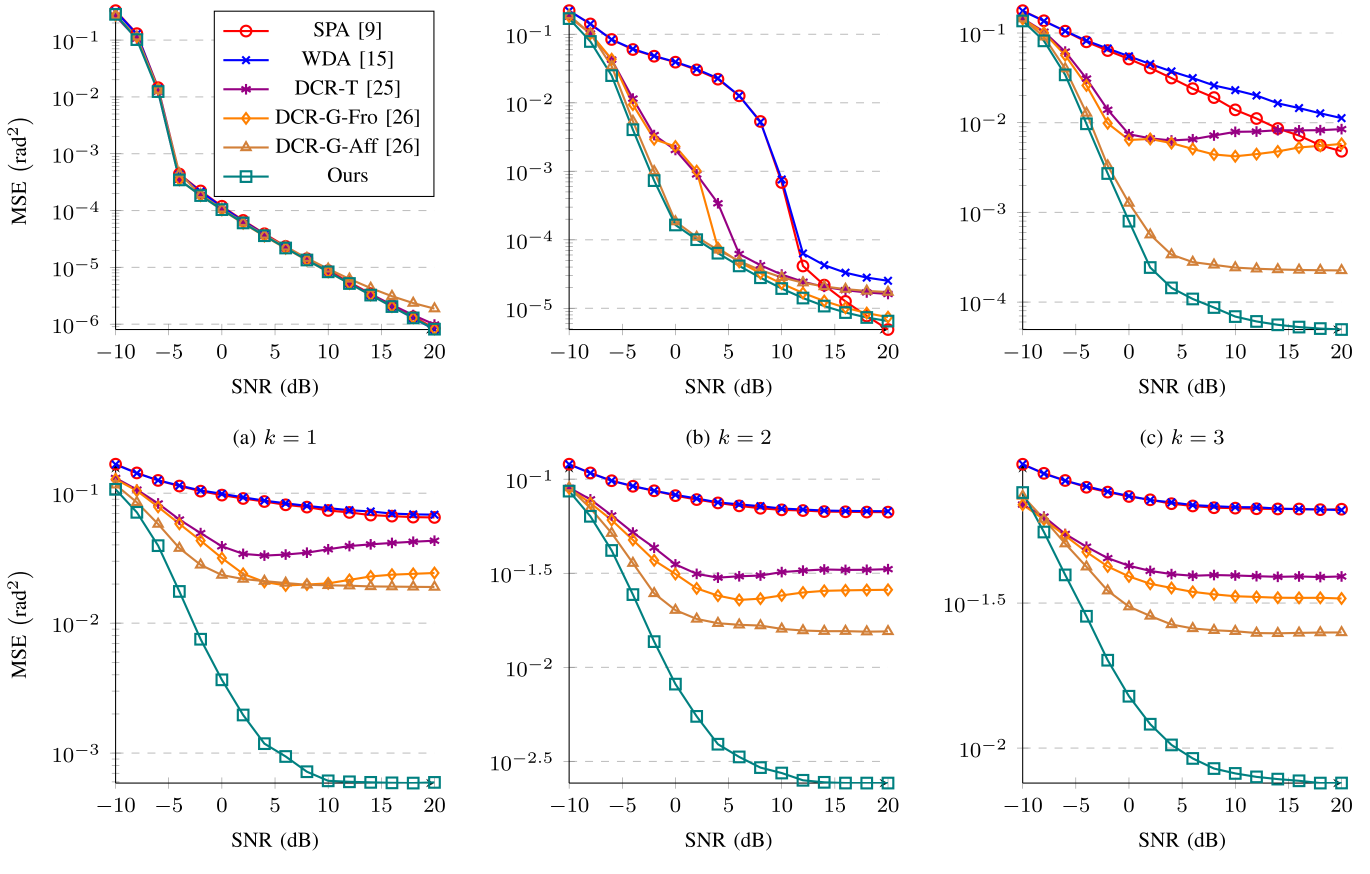
B. Results

1) Superior Performance Over a Wide Range of SNRs: Fig. 3 compares the proposed method with the five baseline approaches in terms of MSE over a wide range of SNRs and number of sources. For $k = 1$, all of the methods have almost the same performance. For $k = 2$, the proposed method is significantly better than SPA and WDA from −10 to 6 dB SNR. In fact, it is uniformly better than WDA from −10 to 20 dB SNR. However, once the SNR goes beyond 14 dB, SPA starts to



(a) $k = 1$ (b) $k = 6$ (c) $k = 9$

Fig. 4. MSE vs. number of snapshots. Although the DNN models are only trained on a single number of snapshots $T = 50$, they are capable of performing well on a wide range of unseen scenarios from $T = 10$ to $T = 100$. Our approach is consistently better than SPA, WDA, and DCR-G-Aff.

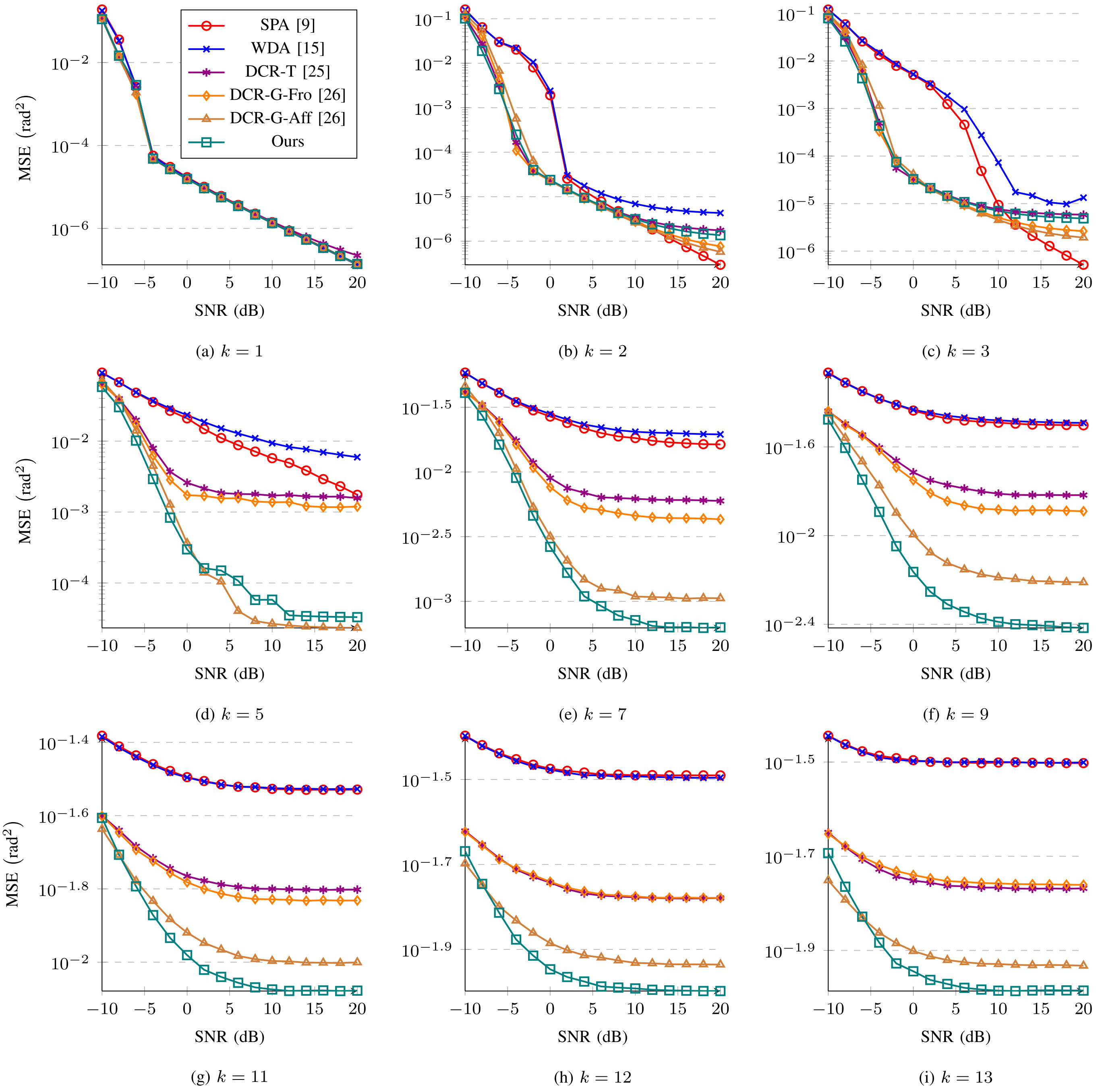


(a) $k = 1$ (b) $k = 2$ (c) $k = 3$

(d) $k = 4$ (e) $k = 5$ (f) $k = 6$

Fig. 5. MSE vs. SNR. $N = 4$. $M = 7$. Our approach is significantly better than all of the baselines when $k > 2$. For $k = 2$, it is better than all of the DNN-based baselines but slightly worse than the SPA at 20 dB SNR. The main results obtained for the 5-element MRA are similar to the 4-element MRA.

outperform the proposed method and the gap seems to become larger as the SNR increases. DCR-T is slightly worse than the proposed method in the high SNR region but the gap of MSE gets larger as SNR increases. With regard to DCR-G-Fro and DCR-G-Aff, their performance is similar to the proposed method. For $k = 3$, the proposed method is better than SPA, WDA, DCR-T, and DCR-G-Fro across almost the entire evaluation range and even superior by orders of magnitude from 0 to 15 dB SNR with respect to SPA, WDA, and DCR-T. DCR-G-Aff is slightly better than the proposed method in the high SNR region but is slightly worse in the low SNR region. Then, for $k \in \{4, 5, 6, 7, 8, 9\}$, the proposed method consistently and



(a) $k = 1$

(b) $k = 2$

(c) $k = 3$

(d) $k = 5$

(e) $k = 7$

(f) $k = 9$

(g) $k = 11$

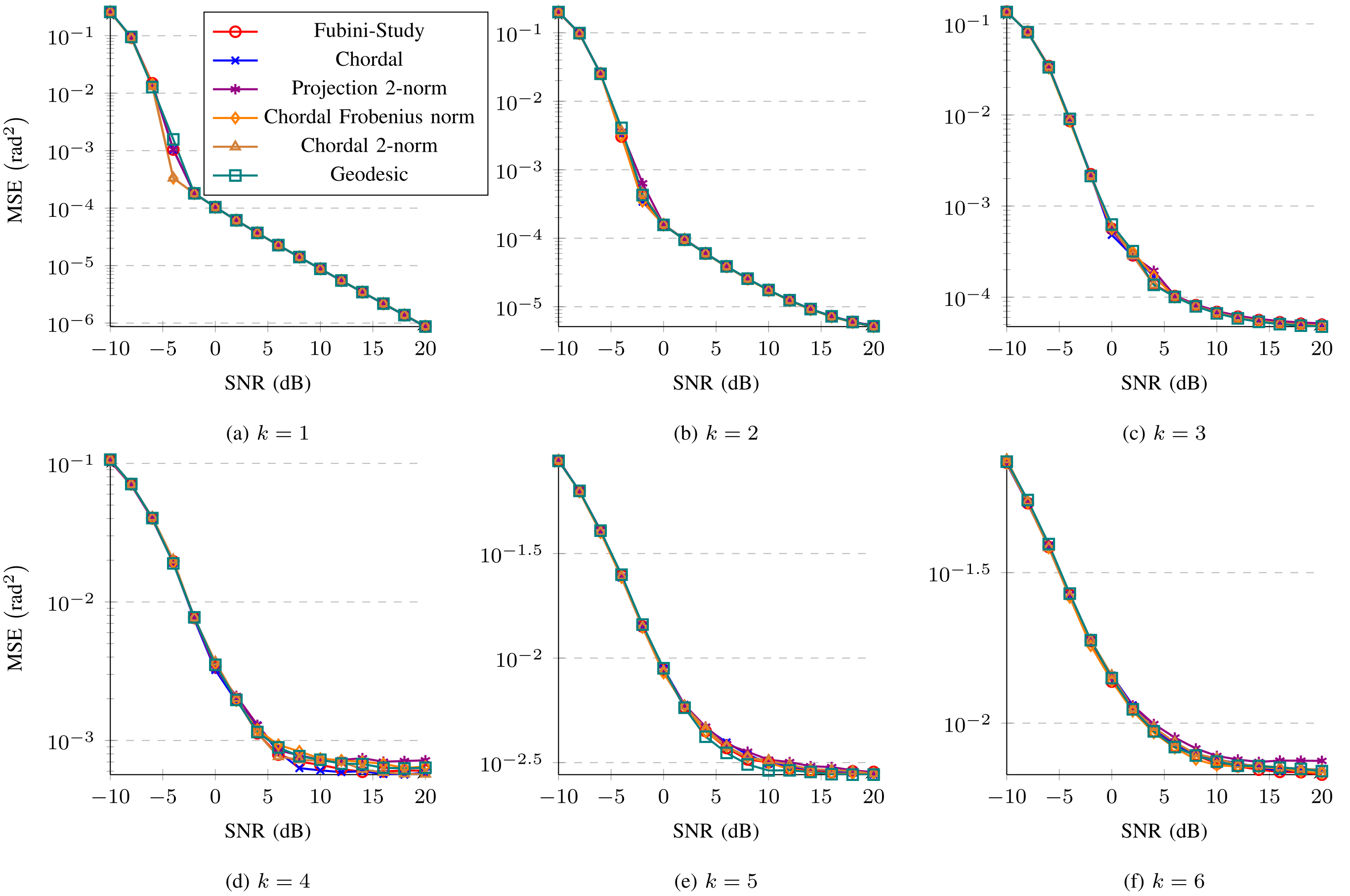
(h) $k = 12$

(i) $k = 13$

Fig. 6. MSE vs. SNR. $N = 6$. $M = 14$. These results, along with Figs. 3 and 5, imply that the proposed method consistently outperforms all of the baselines if $k \geq N$. The proposed method is slightly inferior than DCR-G-Aff at high SNRs when $k < N$.

significantly outperforms SPA and WDA. As for DCR-T and DCR-G-Fro, they are noticeably better than SPA and WDA but significantly inferior than the proposed method. In particular, DCR-G-Aff is the most competitive approach to the proposed method. However, it is still much inferior than our approach. Overall, the proposed method is significantly better than all of the baseline approaches.

2) Performance on Unseen Numbers of Snapshots: Fig. 4 evaluates SPA, WDA, DCR-G-Aff, and the proposed method in terms of MSE in a wide range of numbers of snapshots and sources. We do not include the other baselines here because DCR-G-Aff is significantly better than them according to Fig. 3. For $k = 1$, all of the methods have similar performance. For $k = 6$ and $k = 9$, the proposed method is consistently and significantly better than SPA, WDA, and DCR-G-Aff. More importantly, Fig. 4 also implies that a DNN model trained by the subspace representation learning approach on a specific number of snapshots



(a) $k = 1$

(b) $k = 2$

(c) $k = 3$

(d) $k = 4$

(e) $k = 5$

(f) $k = 6$

Fig. 7. Performance evaluation using different distances in Table I for learning the signal subspace shows nearly identical performance, implying that the choice of distance between subspaces is likely not sensitive to the DoA estimation performance in our methodology.

can perform well across a wide range of unseen numbers of snapshots.

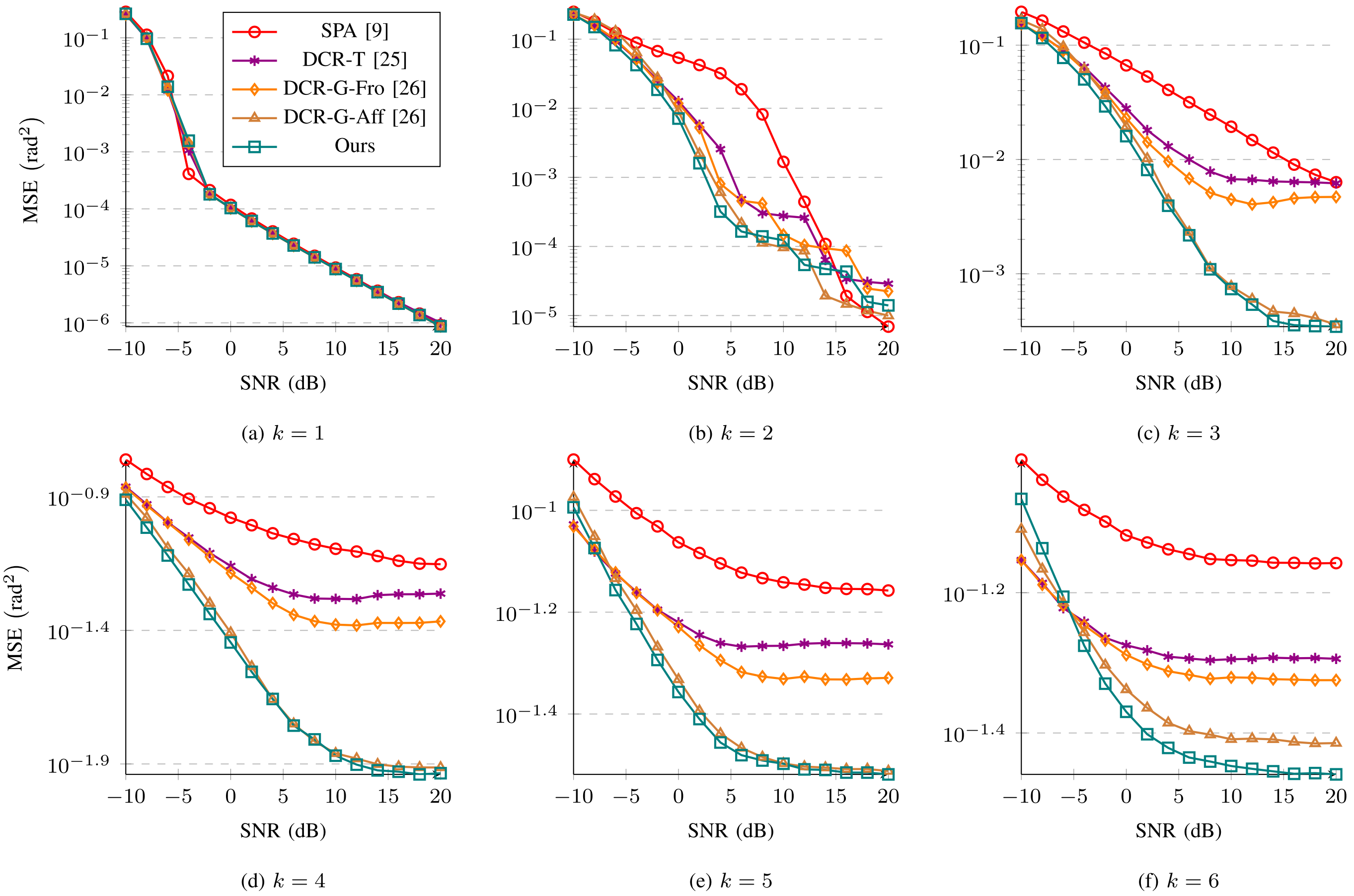
3) On Different SLAs: It is desirable to show that the main conclusions drawn from the 5-element MRA experiments in Section VI-B1 in general hold true for an arbitrary N-element MRA. Here, we demonstrate that this is true for 4-element and 6-element MRAs. Most of the hyperparameters stay the same as the setting in Section VI. We find that $\delta = 10^{-4}$ leads to unstable training in the DCR-G-Aff approach for the case of the 6-element MRA so we increase δ to 10^{-3} in this particular case. Results for the 4-element MRA $\mathcal{S} = \{1, 2, 5, 7\}$ are shown in Fig. 5. Results for the 6-element MRA $\mathcal{S} = \{1, 2, 5, 6, 12, 14\}$ are shown in Fig. 6. All of these results in Figs. 3, 5, and 6 demonstrate that the proposed method outperforms all of the baseline approaches. Furthermore, they seem to suggest our approach is consistently better than all baselines if $k \geq N$. Although we do not include the results on the number of snapshots for the 4-element and 6-element MRAs in this paper (they can be found in [52]), our experiments show that they enjoy the same conclusion drawn from Fig. 4.

4) Other Distances Between Subspaces: Although the geodesic distance, d_k^{Geo}, is the most natural choice for d_k and is used to demonstrate the proposed methodology, other choices for d_k are also possible. To study the effectiveness of subspace representation learning using different distances, we conduct an experiment under the setting of a 4-element MRA, with the same setup as described in Section VI-B3. Fig. 7 shows that models trained using the distances in Table I result in nearly identical performance for DoA estimation. This result aligns with theory, as $d_k^a \to 0$ implies $d_k^b \to 0$ when d_k^a and d_k^b are two different distances listed in Table I. Therefore, we argue that using different distances between subspaces is likely to yield similar performance in our methodology.

5) Random Source Powers: To relax the equal power assumption, new models are trained and evaluated with random source powers satisfying the condition $\frac{\max_i p_i}{\min_j p_j} \leq 10$. Except for the source power assumption, we follow the same setting as the 4-element MRA used in Section VI-B3. Fig. 8 shows that our approach significantly outperforms SPA, DCR-T, and DCR-G-Fro. Although the performance gap between DCR-G-Aff and our approach is greatly reduced compared to Fig. 5, the relative ranking of these methods remains unchanged.

C. Comparison to the Proposed Gridless End-to-End Approach

To answer the question posed in Section V, we use the same WRN-16-8 but replace the final affine layer by $M - 1$ affine



(a) $k = 1$

(b) $k = 2$

(c) $k = 3$

(d) $k = 4$

(e) $k = 5$

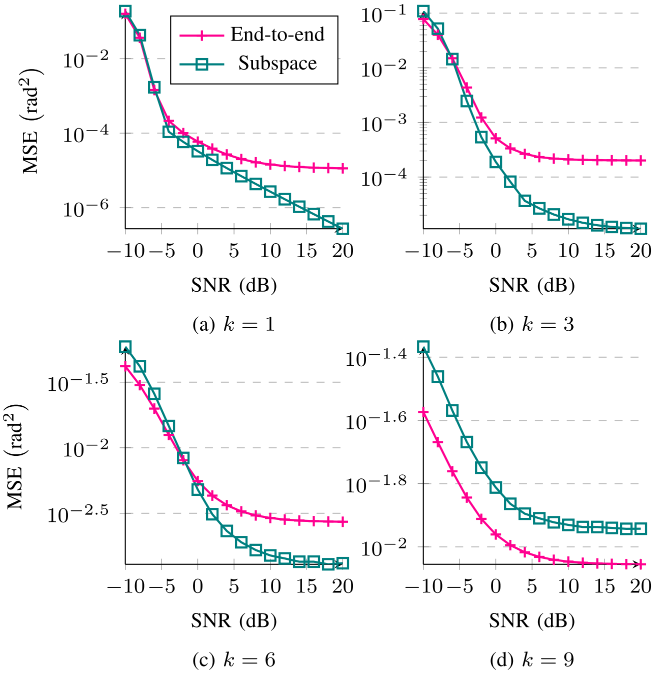
(f) $k = 6$

Fig. 8. Performance evaluation under scenarios with random source powers shows that our approach achieves the best performance overall, but it is only marginally better than DCR-G-Aff. The ranking aligns with the equal power case shown in Fig. 5, although the performance gap between DCR-G-Aff and our approach is much smaller here.

heads whose number of output neurons are $1, 2, 3, \cdots, M-1$, as illustrated in Fig. 1. The squared loss functions of different dimensions are adopted as shown in (27). All of the settings here are the same as the ones described in Section VI-A and VI-A2. The best learning rate is 0.2 according to a simple grid search. Fig. 9 shows that the gridless end-to-end approach tends to saturate its performance earlier than the subspace representation learning approach for $k \in \{1, 3, 6\}$ as the SNR increases. As a result, the subspace representation learning approach shows significantly better performance at high SNRs. However, for $k = 9$, it is consistently worse than the gridless end-to-end approach. Although the gridless end-to-end approach does not have a grid at the output layer, its behavior of hitting an early plateau seems to be similar to grid-based methods that are limited by their grid resolution. Overall, subspace representation learning gives better performance than the gridless end-to-end approach and we can deduce that learning subspace representations is more beneficial than learning angles directly.

D. Robustness to Array Imperfections

With regard to array imperfections, we use the imperfect array manifold introduced by Liu et al. [21]. The exact formulation is given below. Let the degree of imperfections be controlled by a scalar $\rho \in [0, 1]$. A larger ρ makes the



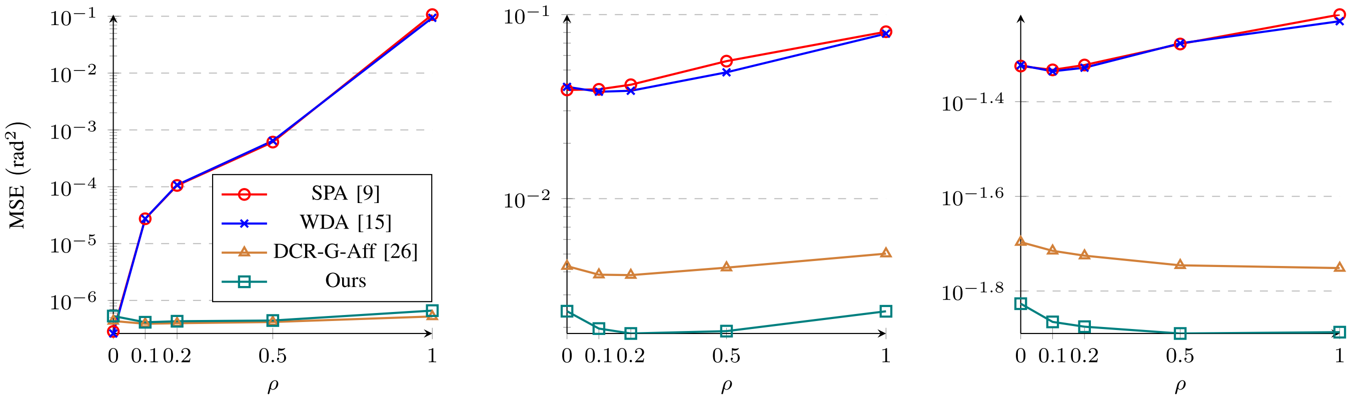
(a) $k = 1$

(b) $k = 3$

(c) $k = 6$

(d) $k = 9$

Fig. 9. MSE vs. SNR. $N = 5$. $M = 10$. For $k \in \{1, 3, 6\}$, the performance of the gridless end-to-end approach saturates at a higher MSE than the subspace representation learning method as the SNR increases. For $k = 9$, the gridless end-to-end approach shows consistently better performance.



(a) $k = 1$

(b) $k = 6$

(c) $k = 9$

Fig. 10. MSE vs. the array imperfection parameter ρ. Note that only one DNN model is trained for our approach. Unlike model-based methods that give significantly worse MSE as ρ increases, our approach is robust to array imperfections without being given ρ or the knowledge about the degree of imperfections.

imperfections more severe and $\rho = 0$ means the array is perfect. Define the following real hyperparameters

$$e_1, \cdots, e_M, g_1, \cdots, g_M, h_1, \cdots, h_M \tag{29}$$

and a complex hyperparameter γ. The array manifold with sensor position errors is given by $\mathbf{a}_\rho(\theta) : [0, \pi] \to \mathbb{C}^M$ such that

$$[\mathbf{a}_\rho(\theta)]_i = e^{j2\pi\left(i-1-\frac{(M-1)}{2}+\rho e_i\right)\frac{d}{\lambda}\cos\theta} \tag{30}$$

for $i \in [M]$. Then, an imperfect array manifold $\tilde{\mathbf{a}}_\rho(\theta)$ of an M-element ULA can be defined by

$$\tilde{\mathbf{a}}_\rho(\theta) = \mathbf{C}_\rho \mathbf{G}_\rho \mathbf{H}_\rho \mathbf{a}_\rho(\theta) \tag{31}$$

where the gain bias is modeled by

$$\mathbf{G}_\rho = \mathbf{I} + \rho \operatorname{diag}(g_1, g_2, \cdots, g_M), \tag{32}$$

the phase bias is modeled by

$$\mathbf{H}_\rho = \operatorname{diag}\left(e^{j\rho h_1}, e^{j\rho h_2}, \cdots, e^{j\rho h_M}\right), \tag{33}$$

and the intersensor mutual coupling is modeled by

$$\mathbf{C}_\rho = \mathbf{I} + \rho \operatorname{Toep}\left(\begin{bmatrix} 0 & \gamma & \gamma^2 & \cdots & \gamma^{M-1} \end{bmatrix}^\mathsf{T}\right). \tag{34}$$

For the hyperparameters, we use $e_1 = 0, e_2 = \cdots = e_6 = -0.2, e_7 = \cdots = e_{10} = 0.2$, $g_1 = 0, g_2 = \cdots = g_6 = 0.2, g_7 = \cdots = g_{10} = -0.2$, and $h_1 = 0, h_2 = \cdots = h_6 = -\frac{1}{6}\pi, h_7 = \cdots = h_{10} = \frac{1}{6}\pi$. To train a model for imperfect arrays, we uniformly select ρ at random on the unit interval $[0, 1]$. To train a model for a perfect array, we use $\rho = 0$.

Fig. 10 shows the MSE in terms of the array imperfection parameter ρ for different numbers of sources. Indeed, a different ρ represents a different array; thus, one can collect a new dataset and then specifically train a new model. However, here, we train a single model on a joint dataset collected from different arrays. The MSE of SPA and WDA both get worse as ρ increases, verifying that both methods suffer from model mismatch and are not robust to array imperfections. Even though no attempts have been made to SPA and WDA to contend with array imperfections, such corrections are nontrivial and require the knowledge of imperfections. On the other hand, the MSE of the proposed method stays at the same level despite the increasing degree of imperfections, implying that the subspace representation learning approach is robust to array imperfections. Although we do not include the results for the 4-element and 6-element MRAs here (they can be found in [52]), our experiments show that they enjoy the same conclusion drawn from Fig. 10.

E. Consistent Rank Sampling

To study the speedup and potential performance regression of consistent rank sampling, two subspace representation learning models are trained with and without consistent rank sampling. This study is conducted on a 4-element MRA with the same setup used in Section VI-B3. The model trained without consistent rank sampling achieves an empirical risk of 0.21312 on the validation set and a training speed of 356.67 seconds per epoch. On the other hand, the model trained with consistent rank sampling achieves an empirical risk of 0.21322 and a training speed of 172.30 seconds per epoch. As a result, consistent rank sampling provides about a 2× training speedup with negligible performance regression. The study is run on an NVIDIA RTX 4090 GPU.

VII. Conclusion

A new methodology learning subspace representations is proposed for robust estimation of more sources than sensors. To learn subspace representations, the codomain of a DNN model, is defined as a union of Grassmannians reflecting signal subspaces of different dimensions. Then, a family of loss functions is proposed as functions of the principal angles between subspaces to ensure rational invariance. In particular, we use geodesic distances on Grassmannians to train a DNN model and prove that it is possible for a ReLU network to approximate signal subspaces. Because a subspace is invariant to the selection of the basis, our methodology expands the solution space of a DNN model compared to existing approaches that learn covariance matrices. In addition, due to its geometry-agnostic nature, our methodology is robust to array imperfections. To study the possibility of bypassing the root-MUSIC algorithm, we propose a gridless end-to-end approach that directly learns a mapping from sample SCMs to DoAs. Numerical results show that subspace representation learning outperforms existing SDP-based approaches including the SPA and WDA, DNN-based covariance matrix reconstruction methods, and the gridless end-to-end approach under the standard assumptions. These results imply that learning subspace representations is more beneficial than learning covariance matrices or angles directly.

Appendix A
Proof of Theorem 1

Proof: Let $\mathcal{I} = \{(i,j) \in [k] \times [k] \mid i \neq j\}$. For every $(i,j) \in \mathcal{I}$, define

$$\mathcal{F}_{i,j} = \left\{ \boldsymbol{\theta} \in [0,\pi]^k \mid \theta_i = \theta_j \right\}. \tag{35}$$

Pick $\delta > 0$ and let μ denote the Lebesgue measure. Because $\mathcal{F}_{i,j}$ is closed and $\mu(\mathcal{F}_{i,j}) = 0$ for every $(i,j) \in \mathcal{I}$, there exists an open set $\mathcal{F}_\delta \supset \bigcup_{(i,j)\in\mathcal{I}} \mathcal{F}_{i,j}$ such that $\mu(\mathcal{F}_\delta) < \delta$. Therefore, $\mathcal{E}_\delta = [0,\pi]^k \setminus \mathcal{F}_\delta$ is compact. Now, note that $\mathbf{A}(\boldsymbol{\theta})$ is a rank-k matrix for every $\boldsymbol{\theta} \in \mathcal{E}_\delta$ due to the Vandermonde structure. It follows that the function $\mathbf{X} \mapsto P_{\mathbf{X}}$ is continuous on $\mathbf{A}(\mathcal{E}_\delta)$. On the other hand, the mapping $\mathbf{R}_\mathcal{S} \mapsto \mathbf{R}_0$ is affine on $\mathbf{A}(\mathcal{E}_\delta)$ since the SLA has no holes in its co-array. As $\mathbf{R}_0 = \mathbf{A}(\boldsymbol{\theta})\mathbf{P}\mathbf{A}^\mathsf{H}(\boldsymbol{\theta})$, we have $P_{\mathbf{R}_0} = P_{\mathbf{A}(\boldsymbol{\theta})}$, implying that $\mathbf{R}_\mathcal{S} \mapsto P_{\mathbf{A}(\boldsymbol{\theta})}$ is continuous on $\mathbf{A}(\mathcal{E}_\delta)$. By Theorem 1 of [31], any continuous piecewise linear function can be represented by a ReLU network. Because the set of continuous piecewise linear functions is dense in the set of continuous functions on any compact subset of $\mathbb{C}^{N\times N}$, it follows that, for every ϵ, there is a ReLU network f such that

$$\sup_{\boldsymbol{\theta}\in\mathcal{E}_\delta} \left\| f\left(\mathbf{R}_\mathcal{S}\right) - P_{\mathbf{A}(\boldsymbol{\theta})} \right\|_F < \epsilon. \tag{36}$$

Note that $\mathbf{R}_\mathcal{S}(\mathcal{E}_\delta)$ is still compact since $\boldsymbol{\theta} \mapsto \mathbf{R}_\mathcal{S}$ is continuous. By Lemma 1,

$$\int_{\mathcal{E}_\delta} d_k^{\text{Geo}}\left(f\left(\mathbf{R}_\mathcal{S}\right), P_{\mathbf{A}(\boldsymbol{\theta})}\right) d\boldsymbol{\theta} < \pi^k \sqrt{k} \sin^{-1}\left(\frac{\epsilon}{\sqrt{2}}\right). \tag{37}$$

As f is continuous and every nonzero orthogonal projection is bounded, there exists $L > 0$ such that $\left\| f\left(\mathbf{R}_\mathcal{S}\right) - P_{\mathbf{A}(\boldsymbol{\theta})} \right\|_F < L$ for every $\boldsymbol{\theta} \in \mathcal{F}_\delta$, which implies

$$\int_{\mathcal{F}_\delta} d_k^{\text{Geo}}\left(f\left(\mathbf{R}_\mathcal{S}\right), P_{\mathbf{A}(\boldsymbol{\theta})}\right) d\boldsymbol{\theta} < \delta \sqrt{k} \sin^{-1}\left(\frac{L}{\sqrt{2}}\right). \tag{38}$$

The claim is proved because both $\delta > 0$ and $\epsilon > 0$ can be arbitrarily small, and $\sin^{-1}(x) \to 0$ as $x \to 0^+$. □

Appendix B
Proof of Lemma 1

Proof: Because there is a one-to-one correspondance between the set of linear subspaces and the set of orthogonal projectors, a distance $d : \text{Gr}(k,M) \times \text{Gr}(k,M) \to [0,\infty)$ between $\mathcal{U}_1 \in \text{Gr}(k,M)$ and $\mathcal{U}_2 \in \text{Gr}(k,M)$ can be defined as

$$d(\mathcal{U}_1, \mathcal{U}_2) = \|P_{\mathcal{U}_1} - P_{\mathcal{U}_2}\|_F \tag{39}$$

where $P_{\mathcal{U}_1}$ and $P_{\mathcal{U}_2}$ are the orthogonal projectors onto $\mathcal{U}_1$ and $\mathcal{U}_2$, respectively. Then, it follows that $d^2(\mathcal{U}_1,\mathcal{U}_2) = 2k - 2\text{tr}\left(P_{\mathcal{U}_1}P_{\mathcal{U}_2}\right)$ which is equivalent to

$$\begin{aligned} 2k - 2\sum_{i=1}^{k} \sigma_i^2\left(P_{\mathcal{U}_1}P_{\mathcal{U}_2}\right) &= 2k - 2\sum_{i=1}^{k} \sigma_i^2\left(\mathbf{U}_1^\mathsf{H}\mathbf{U}_2\right) \\ &= 2k - 2\sum_{i=1}^{k} \cos^2\phi_i \end{aligned} \tag{40}$$

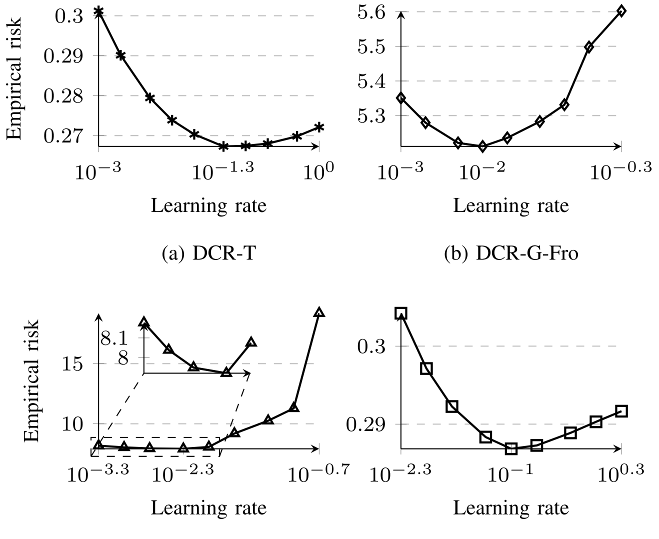
where $\mathbf{U}_1$ and $\mathbf{U}_2$ are matrices whose columns form unitary bases of $\mathcal{U}_1$ and $\mathcal{U}_2$. Therefore, we have

$$\|P_{\mathcal{U}_1} - P_{\mathcal{U}_2}\|_F = \sqrt{2}\left(\sum_{i=1}^{k} \sin^2\phi_i\right)^{\frac{1}{2}} \tag{41}$$

which was shown in [53]. Finally, (41) implies that

$$\phi_i \leq \sin^{-1}\left(\frac{\|P_{\mathcal{U}_1} - P_{\mathcal{U}_2}\|_F}{\sqrt{2}}\right) \tag{42}$$

for every $i \in [k]$. □



(a) DCR-T (b) DCR-G-Fro

(c) DCR-G-Aff (d) Ours

Fig. 11. Search of the best learning rates. Empirical risk on the validation set vs. the maximum learning rate in the one-cycle learning rate scheduler.

Appendix C
Learning Rates

To determine the best learning rates to use in Section VI, a grid search of the best maximum learning rate in the one-cycle learning rate scheduler [49] is performed for each approach. For each $k \in [M-1]$, there are 2×10^6 and 6×10^5 random data points for training and validation, respectively, leading to a training dataset of size $L_{\text{train}} = 9 \times 2 \times 10^6$ and a validation dataset of size $L_{\text{val}} = 9 \times 6 \times 10^5$. Fig. 11 shows that the best learning rates for DCR-T, DCR-G-Fro, DCR-G-Aff, and the proposed approach are 0.05, 0.01, 0.005, and 0.1, respectively. These learning rates are also used to train all the corresponding models in Section VI-B3, VI-B4, VI-B5, VI-C, and VI-E.

Appendix D
Invariance-Aware Loss Functions

There are other loss functions that deviate from covariance reconstruction for gridless DoA estimation [54]. These loss functions are developed based on the scale-invariant signal-to-distortion ratio (SI-SDR) to partially address the primary issue identified in Section IV. Although the new loss functions proposed in [54] underperform subspace representation learning, they provide evidence that loss functions with greater degrees of invariance can achieve better DoA estimation performance.

Acknowledgment

The authors would like to thank Dr. Ching-Hua Lee for his discussions on the gridless end-to-end training and Mr. Rushabha Balaji for his suggestions and comments regarding consistent rank sampling.

Data Availability Statement

Code is available at https://github.com/kjason/SubspaceRepresentationLearning.

References

[1] L. Pisha et al., "A wearable, extensible, open-source platform for hearing healthcare research," *IEEE Access*, vol. 7, pp. 162083–162101, Jul. 2019.

[2] A. Sant and B. D. Rao, "DOA estimation in systems with nonlinearities for mmwave communications," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* Piscataway, NJ, USA: IEEE Press, 2020, pp. 4537–4541.

[3] Y. Liu, H. Chen, and B. Wang, "DOA estimation based on CNN for underwater acoustic array," *Appl. Acoust.*, vol. 172, p. 107594, 2021.

[4] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[5] H. L. Van Trees, *Optimum Array Processing: Part IV of Detection, Estimation, and Modulation Theory*. Hoboken, NJ, USA: John Wiley & Sons, 2002.

[6] P. Sarangi, M. C. Hücümenoğlu, R. Rajamäki, and P. Pal, "Super-resolution with sparse arrays: A nonasymptotic analysis of spatiotemporal trade-offs," *IEEE Trans. Signal Process.*, vol. 71, pp. 4288–4302, 2023.

[7] S. U. Pillai, Y. Bar-Ness, and F. Haber, "A new approach to array geometry for improved spatial spectrum estimation," *Proc. IEEE*, vol. 73, no. 10, pp. 1522–1524, Oct. 1985.

[8] H. Qiao and P. Pal, "On maximum-likelihood methods for localizing more sources than sensors," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 703–706, May 2017.

[9] Z. Yang, L. Xie, and C. Zhang, "A discretization-free sparse and parametric approach for linear array signal processing," *IEEE Trans. Signal Process.*, vol. 62, no. 19, pp. 4959–4973, 2014.

[10] P. Stoica and T. Söderström, "On reparametrization of loss functions used in estimation and the invariance principle," *Signal Process.*, vol. 17, no. 4, pp. 383–387, 1989.

[11] B. Ottersten, P. Stoica, and R. Roy, "Covariance matching estimation techniques for array signal processing applications," *Digit. Signal Process.*, vol. 8, no. 3, pp. 185–210, 1998.

[12] R. R. Pote and B. D. Rao, "Maximum likelihood-based gridless DoA estimation using structured covariance matrix recovery and SBL with grid refinement," *IEEE Trans. Signal Process.*, vol. 71, pp. 802–815, 2023.

[13] G. Tang, B. N. Bhaskar, and B. Recht, "Near minimax line spectral estimation," *IEEE Trans. Inf. Theory*, vol. 61, no. 1, pp. 499–512, Jan. 2014.

[14] Y. Li and Y. Chi, "Off-the-grid line spectrum denoising and estimation with multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 64, no. 5, pp. 1257–1269, May 2015.

[15] M. Wang, Z. Zhang, and A. Nehorai, "Grid-less DOA estimation using sparse linear arrays based on Wasserstein distance," *IEEE Signal Process. Lett.*, vol. 26, no. 6, pp. 838–842, Jun. 2019.

[16] P. Sarangi, M. C. Hücümenoğlu, and P. Pal, "Beyond coarray MUSIC: Harnessing the difference sets of nested arrays with limited snapshots," *IEEE Signal Process. Lett.*, vol. 28, pp. 2172–2176, 2021.

[17] X. Wu, W.-P. Zhu, and J. Yan, "A Toeplitz covariance matrix reconstruction approach for direction-of-arrival estimation," *IEEE Trans. Veh. Technol.*, vol. 66, no. 9, pp. 8223–8237, Sep. 2017.

[18] C. Zhou, Y. Gu, X. Fan, Z. Shi, G. Mao, and Y. D. Zhang, "Direction-of-arrival estimation for coprime array via virtual array interpolation," *IEEE Trans. Signal Process.*, vol. 66, no. 22, pp. 5956–5971, 2018.

[19] P. Stoica, P. Babu, and J. Li, "SPICE: A sparse covariance-based estimation method for array processing," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 629–638, Feb. 2010.

[20] P. Stoica and P. Babu, "SPICE and LIKES: Two hyperparameter-free methods for sparse-parameter estimation," *Signal Process.*, vol. 92, no. 7, pp. 1580–1590, 2012.

[21] Z.-M. Liu, C. Zhang, and S. Y. Philip, "Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections," *IEEE Trans. Antennas Propag.*, vol. 66, no. 12, pp. 7315–7327, Dec. 2018.

[22] G. K. Papageorgiou, M. Sellathurai, and Y. C. Eldar, "Deep networks for direction-of-arrival estimation in low SNR," *IEEE Trans. Signal Process.*, vol. 69, pp. 3714–3729, 2021.

[23] A. Barthelme and W. Utschick, "A machine learning approach to DoA estimation and model order selection for antenna arrays with subarray sampling," *IEEE Trans. Signal Process.*, vol. 69, pp. 3075–3087, 2021.

[24] K.-L. Chen, C.-H. Lee, B. D. Rao, and H. Garudadri, "A DNN based normalized time-frequency weighted criterion for robust wideband DoA estimation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* Piscataway, NJ, USA: IEEE Press, 2023, pp. 1–5.

[25] X. Wu, X. Yang, X. Jia, and F. Tian, "A gridless DOA estimation method based on convolutional neural network with Toeplitz prior," *IEEE Signal Process. Lett.*, vol. 29, pp. 1247–1251, 2022.

[26] A. Barthelme and W. Utschick, "DoA estimation using neural network-based covariance matrix reconstruction," *IEEE Signal Process. Lett.*, vol. 28, pp. 783–787, 2021.

[27] N. J. Higham, *Functions of Matrices: Theory and Computation*. Society for Industrial and Applied Mathematics, 2008.

[28] P. Pal and P. P. Vaidyanathan, "Nested arrays: A novel approach to array processing with enhanced degrees of freedom," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4167–4181, Aug. 2010.

[29] A. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *Proc. Int. Conf. Acoust., Speech, Signal Process.* Piscataway, NJ, USA: IEEE Press, 1983, pp. 336–339.

[30] B. D. Rao and K. S. Hari, "Performance analysis of root-music," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 12, pp. 1939–1949, Dec. 1989.

[31] K.-L. Chen, H. Garudadri, and B. D. Rao, "Improved bounds on neural complexity for representing piecewise linear functions," *Adv. Neur. Inf. Process. Syst.*, vol. 35, 2022, pp. 7167–7180.

[32] P. Stoica, P. Babu, and J. Li, "New method of sparse parameter estimation in separable models and its use for spectral analysis of irregularly sampled data," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 35–47, Jan. 2010.

[33] R. Bhatia, T. Jain, and Y. Lim, "On the Bures–Wasserstein distance between positive definite matrices," *Expositiones Mathematicae*, vol. 37, no. 2, pp. 165–191, 2019.

[34] K.-C. Toh, M. J. Todd, and R. H. Tütüncü, "SDPT3—A MATLAB software package for semidefinite programming, version 1.3," *Optim. Methods Softw.*, vol. 11, no. 1–4, pp. 545–581, 1999.

[35] R. Bhatia, *Positive Definite Matrices.* Princeton, NJ, USA: Princeton University Press, 2007.

[36] Y.-C. Wong, "Differential geometry of grassmann manifolds," *Proc. Nat. Acad. Sciences*, vol. 57, no. 3, pp. 589–594, Mar. 1967.

[37] Å. Björck and G. H. Golub, "Numerical methods for computing angles between linear subspaces," *Math. Comput.*, vol. 27, no. 123, pp. 579–594, 1973.

[38] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 2, pp. 303–353, Feb. 1998.

[39] A. Barg and D. Y. Nogin, "Bounds on packings of spheres in the grassmann manifold," *IEEE Trans. Inf. Theory*, vol. 48, no. 9, pp. 2450–2454, Sep. 2002.

[40] J. Hamm and D. D. Lee, “Grassmann discriminant analysis: A unifying view on subspace-based learning,” in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 376–383.

[41] K. Ye and L.-H. Lim, “Schubert varieties and distances between subspaces of different dimensions,” *SIAM J. Matrix Anal. Appl.*, vol. 37, no. 3, pp. 1176–1197, Mar. 2016.

[42] G. H. Hardy, J. E. Littlewood, and G. Pólya, *Inequalities*. Cambridge, U.K.: Cambridge University Press, 1934.

[43] M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming, version 2.1,” https://cvxr.com/cvx, Mar. 2014.

[44] M. Grant and S. Boyd, “Graph implementations for nonsmooth convex programs,” in *Recent Advances in Learning and Control* (Lecture Notes in Control and Information Sciences), V. Blondel, S. Boyd, and H. Kimura, Eds. New York: Springer-Verlag Limited, 2008, pp. 95–110.

[45] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. Conf. Comput. Vision Pattern Recognit.* Piscataway, NJ, USA: IEEE Press, 2016, pp. 770–778.

[46] S. Zagoruyko and N. Komodakis, “Wide residual networks,” in *Proc. Brit. Mach. Vision Conf.* BMVA Press, 2016, pp. 87.1–87.12.

[47] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” in *Proc. Eur. Conf. Comput. Vision*. New York: Springer, 2016, pp. 630–645.

[48] K.-L. Chen, C.-H. Lee, H. Garudadri, and B. D. Rao, “ResNEsts and DenseNEsts: Block-based DNN models with improved representation guarantees,” *Adv. Neur. Inf. Process. Syst.*, vol. 34, 2021, pp. 3413–3424.

[49] L. N. Smith and N. Topin, “Super-convergence: Very fast training of neural networks using large learning rates,” in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, vol. 11006. SPIE, 2019, pp. 369–386.

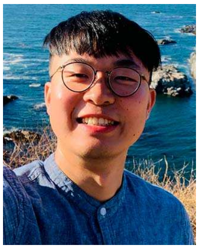
[50] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in *Proc. Int. Conf. Comput. Vision*. Piscataway, NJ, USA: IEEE Press, 2015, pp. 1026–1034.

[51] A. Paszke et al., “PyTorch: An imperative style, high-performance deep learning library,” in *Adv. Neur. Inf. Process. Syst.*, vol. 32, 2019.

[52] K.-L. Chen, “Deep learning with estimation and complexity guarantees for signal processing,” Ph.D. dissertation, University of California, San Diego, 2024.

[53] G. W. Stewart, “Perturbation theory for the singular value decomposition,” in *SVD and Signal Processing II, Algorithms, Analysis and Applications*, R. J. Vaccaro, Ed., New York, NY, USA: Elsevier Science Publishers, 1991, pp. 99–109.

[54] K.-L. Chen and B. D. Rao, “A comparative study of invariance-aware loss functions for deep learning-based gridless direction-of-arrival estimation,” in *Proc. Int. Conf. Acoust., Speech, Signal Process.* Piscataway, NJ, USA: IEEE Press, 2025.



Kuan-Lin Chen (Member, IEEE) received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 2016, and the M.S. and Ph.D. degrees from the University of California, San Diego (UCSD), La Jolla, CA, USA, in 2019 and 2024, respectively. After his graduate studies, he joined Apple as a Machine Learning Research Engineer. He was a recipient of the Qualcomm Innovation Fellowship, in 2022, the Innovative Research Grants Award from the Kavli Institute for Brain and Mind, in 2021, the IEEE Signal Processing Society Scholarship, in 2023, the NeurIPS Scholar Award, in 2022, the ICASSP Rising Stars in Signal Processing, in 2023, and the Best TA Award from the Department of Electrical and Computer Engineering, UCSD, in 2024. His research interests span several areas of machine learning and signal processing, including neural networks, deep learning, array processing, and speech processing.

Bhaskar D. Rao (Life Fellow, IEEE) received the B.Tech. degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in 1979, and the M.S. and Ph.D. degrees from the University of Southern California, Los Angeles, in 1981 and 1983, respectively. Since 1983, he has been teaching and conducting research with the University of California, San Diego, La Jolla, where he is currently a Professor Emeritus and Distinguished Professor of the Graduate Division in the Electrical and Computer Engineering department. He has also been the holder of the Ericsson Endowed Chair in Wireless Access Networks and Distinguished Professor and the Director of the Center for Wireless Communications, from 2008 to 2011. His research interests are in the areas of statistical signal processing, estimation theory, optimization theory, and machine learning, with applications to digital communications, speech signal processing, and biomedical signal processing. He is a pioneer in the theory and use of sparsity in signal processing applications. Since co-authoring the first paper on the seminal FOCUSS algorithm in 1992, he has been driving the field of sparsity forward, including co-organizing the first special session on sparsity at ICASSP 1998 entitled “SPEC-DSP: Signal Processing with Sparseness Constraint.” His work has received several paper awards, including the 2012 Signal Processing Society (SPS) Best Paper Award for the paper “An Empirical Bayesian Strategy for Solving the Simultaneous Sparse Approximation Problem,” with David P. Wipf and the Stephen O. Rice Prize Paper Award in the field of communication systems for the paper “Network Duality for Multiuser MIMO Beamforming Networks and Applications,” with B. Song and R. L. Cruz. He was elected as a fellow of IEEE in 2000 for his contributions to the statistical analysis of subspace algorithms for harmonic retrieval. He received the IEEE Signal Processing Society Technical Achievement Award, in 2016 and was the recipient of the 2023 IEEE SPS Norbert Wiener Society Award. He has been a member of the Statistical Signal and Array Processing Technical Committee, the Signal Processing Theory and Methods Technical Committee, the Communications Technical Committee of the IEEE Signal Processing Society, SPS Fellow Evaluation Committee, from 2023 to 2024 and was the Chair of the Machine Learning for Signal Processing Technical Committee, from 2019 to 2020.